

(51)Int.Cl.<sup>6</sup>  
G 0 6 F 3/14識別記号  
3 4 0F I  
G 0 6 F 3/14技術表示箇所  
3 4 0 A

## 審査請求 未請求 請求項の数1 O.L (全44頁)

(21)出願番号 特願平8-288851  
 (22)出願日 平成8年(1996)10月30日  
 (31)優先権主張番号 552663  
 (32)優先日 1995年11月3日  
 (33)優先権主張国 米国(US)

(71)出願人 590000798  
 ゼロックス コーポレイション  
 XEROX CORPORATION  
 アメリカ合衆国 ニューヨーク州 14644  
 ロチェスター ゼロックス スクエア  
 (番地なし)  
 (72)発明者 トーマス・ピー・モラン  
 アメリカ合衆国 カリフォルニア州  
 94301 パロアルト グリーンウッドアベ  
 ニュー 1037  
 (74)代理人 弁理士 小堀 益 (外1名)

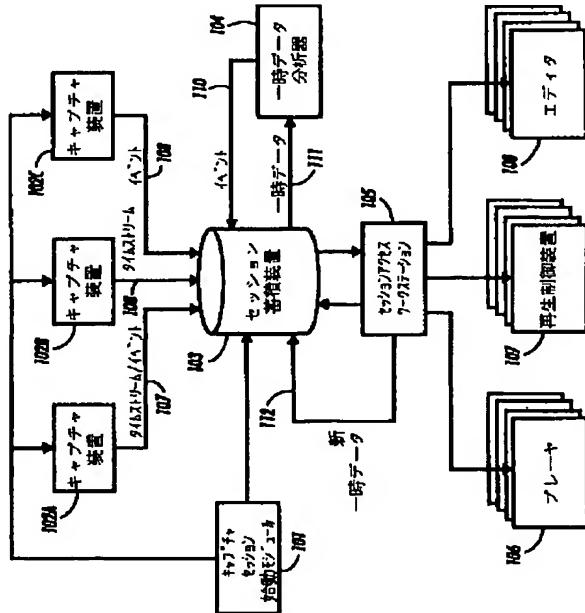
最終頁に続く

## (54)【発明の名称】 コンピュータ制御型ディスプレイシステム

## (57)【要約】

【課題】 コンピュータ制御型ディスプレイシステムにおいて、ミーティングのような共同作業を表現する一時的データの再生を制御するためのユーザインターフェースを提供する。

【解決手段】 ディスプレイの第1のウィンドウのディスプレイのための共有表現メディアとの相互作用を表現するタイムストリームからディスプレイ情報を発生させるための共有表現メディアエミュレーション回路と、ディスプレイの第2のウィンドウのタイムラインにビジュアルインジケータを表示するための表示情報を発生させるためのタイムライン発生回路と、第1或いは第2のウィンドウのいずれかでのユーザ相互作用を検出し、共有表現メディアエミュレーション回路とタイムライン発生回路の各々に対する制御信号を発生させ、ユーザの相互作用に応じて第1と第2のウィンドウを更新するユーザインターフェース制御回路を備えたシステム。



## 【特許請求の範囲】

【請求項1】複数のメディアキャプチャ装置で獲得されたセッションの再生を制御するためのコンピュータ制御型ディスプレイシステムであって、前記複数のメディアキャプチャ装置の少なくとも一つが共有表現メディアであり、前記共有表現メディアがグラフィックオブジェクトを生成して表示するものであり、前記コンピュータ制御型ディスプレイシステムが前記セッションを再生するための一つ以上のプレーヤに連結されており、前記コンピュータ制御型ディスプレイシステムが、

ディスプレイと、

ユーザが前記コンピュータ制御型ディスプレイシステムと相互に作用することを可能にするためのカーソル制御装置と、

前記獲得されたセッションの一時的数据を受け取るためのセッション入力手段であって、前記一時的数据がタイムストリームとイベントから成るものと、前記セッション入力手段に連結された処理回路であって、前記処理回路が前記ディスプレイに表示されたウィンドウのための表示情報を発生させるものとからなり、前記処理回路が、

前記ディスプレイの第1のウィンドウのディスプレイのための前記共有表現メディアとの相互作用を表現するタイムストリームからディスプレイ情報を発生させるための共有表現メディアエミュレーション回路であって、前記第1のウィンドウが、前記共有表現メディアでマーキングの一時的な順序を視覚的に表示するためのプレーヤとして、また、前記セッションの再生を制御するためのユーザインターフェースとして機能するものと、

前記ディスプレイの第2のウィンドウのタイムラインにビジュアルインジケータを表示するための表示情報を発生させるためのタイムライン発生回路であって、前記ビジュアルインジケータの各々がイベントに対応して前記セッションへの一時的なインデックスを表すものと、前記第1のウィンドウ或いは前記第2のウィンドウのいずれかでのユーザ相互作用を検出し、前記共有表現メディアエミュレーション回路と前記タイムライン発生回路の各々に対する制御信号を発生させ、前記ユーザの相互作用に応じて前記第1のウィンドウと前記第2のウィンドウを更新するユーザインターフェース制御回路とから成るコンピュータ制御型ディスプレイシステム。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】本発明は、セッションのマルチメディアキャプチャ、再生、及び、編集の分野に関する。

## 【0002】

【従来の技術】グループの大部分の仕事は、構造化されたミーティングのような整理されたセッティングが行われている場合でさえ、会話による言葉のやりとりのよう

カジュアルな相互作用の形態を採っており、これにより、グループは、彼らの仕事の技術、プロセス、関係がある様相を共通に理解することができる。このカジュアルな活動は、そのような活動の結果に焦点が合っており、グループがどのようにそれらの結果に達したかということについては殆ど無視するコンピュータ使用のツールによっては十分にはサポートされてはいない。更に、そのような情報を収集する試みは、かれらの自然な仕事の慣例をサポートするより、むしろ、活動を公式なものとして、参加者を情報収集ツールに都合が良い方法に従わせる結果となっている。

【0003】ミーティングのオーディオ、ビデオ、及び、コンピュータ記録を収集することは、グループプロセスの十分な検討可能な記録を提供する。残念ながら、そのような記録の利点は、しばしば必要とされたシーケンシャルなアクセスのために扱い難い傾向があった。そのような情報のデジタル記録の出現で、そのような情報に即時にアクセスすることが可能である。しかしながら、即時アクセスは、行くべき場所が判っているときのみ有用である。従って、意味を持つ要素かイベントに基づいて、そのような記録にインデックスを付けることは、必要であり望ましいことである。

【0004】インデックスを付けることへの一つのアプローチは、リアルタイムノートテーキング (Real-Time note taking) と称される。そして、そこで、人は、ノートの種々の個々の要素にタイムスタンプするシステムを使用して、ミーティングの間にノートを取る。これは、ミーティングの記録への理解可能なインデックスを提供する。ノートの任意の要素を単に選択することによって、システムはノート要素のタイムスタンプに対応するAV記録の部分を、検索して再生することができる。

【0005】最も簡単な例は、文の書写システムであり、そこでは、ユーザは、ミーティングが進むに従ってテキストを単にタイプするだけである（テキストは、発言されたことの文字通りの書写或いはミーティングに関する任意のノートとすることができる）。ノートは、テキストのストリングから成り、タイムスタンプされたノート要素は、テキストの個々の文字である。ノートテーキングが完成した後に、ユーザは、文字を選択することができ、ミーティング記録の対応する部分が検索されて再生されることになる。

【0006】ノートテーキングシステムは、ユーザが視覚的な表現（すなわちノート）を構成するのを可能にする。そして、その可視の要素はミーティング記録へのインデックスとして機能を果たす。各々のインデックス要素は、それと関連づけられた時間を有する。この時間は、ユーザがそれを生成した時間である。各々のインデックス要素が可視なので、ユーザは、単に指示によってそれにそれを選択することができ、また、それは関連づ

けられた時間有するので、ミーティング記録へアドレスを得ることができる。

【0007】ノートテーキングシステムは、視覚的な表現の構造が、イベントを生成する要素のシーケンスのみから成っている限りは良好に動作する。表現がミーティングのコースの間に編集されるのを許されると、すなわち、要素変更及び要素削除イベントも同様に存在すると、問題が起こる。問題の原因は、インデックスが可視の要素であって、イベントではないことである。

#### 【0008】

【発明が解決しようとする課題】ノートテーキングによるアプローチの不利な点は、人がノートテーキングの仕事に専念しなければならないことである。従って、人を、自然なミーティング活動それ自身の副作用であるノートテーキングの仕事に専念させないで、ミーティング記録中へインデックスを生成することは望ましい。

#### 【0009】

【課題を解決するための手段】本発明は、ミーティングのような共同作業を表現する一時的データの再生を制御するためのユーザインターフェースを目指すものである。そのような一時的データは、一つ或いはそれ以上のキャプチャ装置（例えば、オーディオの記録或いは電子ホワイトボード）によって獲得される。一時的データは、タイムストリームとイベントから成る。タイムストリームは、ミーティングの或る記録可能な局面を表現するセットのタイムスタンプされたデータである。オーディオ及びビデオの記録、電子ホワイトボードの上の行動の蓄積された履歴は、タイムストリームの例である。イベントは、タイムストリームの間の出来事である。イベントは、共同作業の間の1点またはスパンへの直接アクセスを提供するインデックスを生成するために使われる。タイムストリームは、イベントを固有に定義することができ、または、イベントを識別するために分析される場合がある。イベント情報は、タイムスタンプ、イベント型、及び、イベントのインスタンスの種々の特性から成る。

【0010】セッションの再生は、セッションアクセス装置の制御の下で行なわれる。タイムストリームを再生するための複数のプレーヤが、セッションアクセス装置へ連結された。セッションアクセス装置は、セッション再生を制御するためのユーザインターフェースを生成するためにイベント情報を利用する。ユーザインターフェースは、複数のウィンドウから成る。各々のウィンドウは、プレーヤ、再生制御装置、または、エディタ（これは、ユーザが追加のタイムストリーム情報を生成することを許容する）を表現することができる。本発明においては、二つのウィンドウがコヒーレントな共働作用するやり方で使われる。第1のウィンドウは、タイムストリームを再生するためのプレーヤとして機能を果たし、電子ホワイトボードは同様にセッションの再生を制御する

ために使われる。第2のウィンドウは、タイムラインメタファーを利用する再生制御装置として機能を果たす。第1のウィンドウにおいては、電子ホワイトボード表現の上の各々のマーキングは、一つ或いはそれ以上のイベント（例えば、イベント生成、イベント編集、イベント移動、或いは、イベント削除）と関連付けられる。マーキングの選択は、再生時間が関連づけられたイベントの一つの時間へ動かされることを可能にする。選択された時、イベントは第2のウィンドウのタイムラインの上で表示される。逆に、タイムラインウィンドウのビジュアルインジケータが選択されるとき、第1のウィンドウの対応するオブジェクトがハイライトされる。そのような首尾一貫性は、興味ある情報を含むであろうセッション中の点に、ユーザが正確に目標を定めることを可能にする。

#### 【0011】

【発明の実施の形態】図1は、本発明の現時点での好適な実施態様における一時的データの獲得及び再生のためのシステムのブロック図である。図2は、本発明の現時点での好適な実施態様において利用することができるセッションアクセスワークステーションのブロック図である。図3は、本発明の現時点での好適な実施態様のセッションアクセスワークステーションの動作の単純化された疑似状態である。図4は、本発明の現時点での好適な実施態様のセッションアクセスワークステーションに連結されるディスプレイのビジュアルユーザインターフェースの図である。図5は、本発明の現時点での好適な実施態様のセッションのためのタイムラインユーザインターフェースディスプレイの図である。図6は、概観時間トラックディスプレイエリアと焦点時間トラックディスプレイエリアの関係をハイライトするタイムラインインターフェースの単純化された図である。図7は、図6の焦点バーが引き伸ばされこれに従って焦点時間トラックディスプレイエリアの寸法が変えられている動作の結果を示す図である。図8は、二つのセッションの再生を制御するためのタイムラインユーザインターフェースディスプレイシステムの図である。図9は、二つの不連続な焦点セグメントが概観時間トラックエリアに表示され、焦点時間トラックエリアが各々の焦点セグメントの詳細を含むタイムラインインターフェースの図である。図10は、図9の焦点時間トラックエリアのサイズ修正の結果を示すタイムラインインターフェースの図である。図11は、LiveBoardウィンドウの選択動作の結果として生じたトラックセクションを含む複数のキャプチャ装置で記録したセッションの再生のためのタイムラインインターフェースの例である。図12は、一つのトラックの上で図11のオーディオのトラックを併合した結果を示すタイムラインユーザインターフェースディスプレイの更なる図である。図13は、ノートトラックの上にノートを開けた結果を示す図11のタイムラインユ

ーザインターフェースディスプレイの更なる図である。図14は、図11のタイムラインユーザインターフェースを生成するのに使用した同じセッションについてのミーティングプレーヤの図である。図15は、本発明の現時点での好適な実施態様におけるLiveBoardウインドウの図である。図16は、LiveBoardウインドウとタイムラインインターフェースの間の相関性の図である。すなわち、グラフィックオブジェクトからタイムラインへのインデックス付けである。図17は、Liveboardウインドウとタイムラインインターフェースの間の相関性の図である。すなわち、LiveBoardウインドウとタイムラインインターフェースの両方が同時に同じものを反映する同期化されたディスプレイ状態である。図18は、Liveboardウインドウとタイムラインインターフェースの間の相関性の図である。すなわち、一時的なものから空間へのインデックス付けである。図19は、LiveBoardウインドウとタイムラインインターフェースの間の相関性の図である。すなわち、グラフィックオブジェクトにつき多数のイベントである。図20は、LiveBoardウインドウとタイムラインインターフェースの間の相関性の図である。すなわち、オブジェクトエリアの位置選定によってインデックスを付ける。図21及び図22は、LiveBoardウインドウが編集された前後の“バウンシングボール”再生モードにおけるLiveBoardウインドウの例を提供する。

【0012】本発明は、ミーティングのような共同作業、及び、他の非共同作業の記録の再生の提供する。

【0013】本発明の目的は、作業そのものの自然な副産物である共同作業のミーティング記録へインデックスを生成することである。これは、人々が記録されたセッションの一部への直接アクセスを得る直観的な方法を提供する。話者の変更、ホワイトボードへの書き込み及びマーキング操作、ノートを取ることのようなイベントは、有益なインデックスポイントを提供するミーティングの自然な副産物の例である。

【0014】本発明のシステムは、ソフトウェアに基づいたタイムストリームアーキテクチャ及びシステムアーキテクチャを参照して述べられる。タイムストリームアーキテクチャは、現時点での好適な実施態様が実行されるソフトウェアフレームワークを述べる。システムアーキテクチャは、本発明の機能の構成要素を述べる。

【0015】タイムストリーム及びシステムアーキテクチャの記述で使われる用語は、ここで定義される。

【0016】タイムストリーム(Timestream)は、キャプチャ装置で獲得されたタイムスタンプされたメディアデータを意味し、プレーヤーで再生することができる。

【0017】イベント(Event)は、獲得される活動の間の或る点または間隔で発生する出来事、例え

ば、話者の交代或いはホワイトボードへの書き込みである。

【0018】一時的データ(Temporal Data)は、タイムストリームとイベントの結合を意味する。

【0019】セッション(Sessions)は、ミーティングのように密着して獲得された活動を表現する一時的データの収集物に名前が付けられる。

【0020】プレーヤ(Players)は、その元の形態で、または、タイムストリームのダイナミックスを示す形態で、タイムストリームを再生することができる。

【0021】エディタ(Editors)は、ビジュアルインジケータを変化させることができる、或いは、イベントとタイムストリームをセッションに付加することができるユーザインターフェースを意味する。

【0022】再生制御装置(Playback Controllers)は、セッションの再生が制御されるユーザインターフェースを意味する。

【0023】キャプチャ装置(Capture Devices)は、一時的データを獲得して蓄積する装置である。異なった型のキャプチャ装置は、セッションの異なったメディア(例えば、オーディオ、ビデオ、ホワイトボード上の書き込み)を獲得する。

【0024】タイムストリームアーキテクチャは、調和した記録と再生のための拡張性があるセットのツールの使用をサポートする。これは、特定のマシンの一つのアドレス空間に存在するが、ネットワーク内の同一の或いは他のマシンの同一の或いは異なったアドレス空間に存在する“代理”オブジェクトを有するオブジェクトを、ユーザが容易に構築することを可能にする分散型オブジェクトシステムに基づいている。この分散型オブジェクトシステムは、言語間統一(Inter-Language Unification)(ILU)と称され、Janssenによる“ILUマニュアル”、ゼロックス技術報告書、ISTL-CSA-94-01-02、1994年1月に述べられている。タイムストリームアーキテクチャの基本的なデータ構成要素は、イベント、タイムストリーム、及び、セッションである。

【0025】イベントは、開始時間、期間('ポイント'イベントについては多分ゼロ)、型、及び、型に依存する可能のある特徴からなる。(例えば、LiveBoardページイベントは、'ページ数'と'ページ表題'特性を有するかもしれない。)イベントオブジェクトへのインターフェースは、以下になる。

```
TYPE Event=CLASS
SUPERCLASS PersistentObject
METHODS
```

```

getType() : String,
getStartTime() : Time,
getDuration() : Time
END
【0026】ここで述べられたインターフェース仕様
は、上述したILUシステムの指定言語のスタイルをし
ているが、文字通りそのままであることを意図するもの
ではなく、オブジェクト指向プログラミングを熟知して
いる者によって、容易に翻訳することができる。データ
型ストリングと時間の意味は自明である。付加的に、P
ersistentObjectのサブクラスとして、
イベントは以下の方法を受け継ぐ
getPropertyNames() : String
Sequence
setProperty(name: String, v
alue: String)
getProperty(name: String) :
String
【0027】イベントは、以下に述べたように、セッ
ションオブジェクトへのコールによって生成される。
【0028】タイムストリームオブジェクトは、所定の
メディア型のために蓄積されたデータを参照するため
に使用される。一般的に言えば、タイムストリームと関連
したデータは、記録されたメディアの時間に基づいたア

```

```

TYPE Timestream=CLASS
SUPERCLASS PersistentObject END
METHODS
mediaType() : String,           e. g. audio
format() : String,             e. g. sun au
startTime() : Time,
duration() : Time,
END

```

【0029】セッションは、イベント、タイムストリーム、及び、他のセッションを含むかもしれない階層構造の収集物である。用語'セッション'のこの使用は、時々遭遇するものよりも広義であり、例えば、所定のミーティング或いは討論と関連するデータの狭い意味の'セッション'は特別な場合であることに注意すべきである。セッションは、本質的にタイムストリームデータを組織するための汎用の簿記機構である。セッションオブジェクトへのインターフェースは、以下のようなものである。

```

TYPE Session = CLASS
SUPERCLASS PersistentObject END
METHODS
getTimestreamByName(name:
String) : Timestream,
getAllTimestreams() : Times
TYPE Player=CLASS

```

クセスを可能にする。記録されたオーディオのように、多くのデジタルメディアの関しては、これは、イベントと見なすことができたデータ時間サンプルデータから成るけれども、大量で大きくメディア（また、装置にも）に依存するサンプルされたデータの翻訳は、タイムストリームを'不透明'と見ることを一層実際的にする。すなわち、タイムストリームデータは、タイムストリームのメディアとフォーマットと関連している特別なツールによってのみ正常に操作される。特に、データは、レコーダオブジェクトによって生成され、プレーヤオブジェクトによって再生される。同様に、イベントを作り出すために、メディアの特定のツールによって分析することもできる。（例えば話者セグメント、沈黙／中断、場面変化など）タイムストリームオブジェクトを不透明として取り扱うことにより、アーキテクチャがアナログメディアに適応することができることに注意すべきである。（例えば、レコーダとプレーヤは、コンピュータソフトウェアで制御されるビデオテープレコーダかもしれません、タイムストリームは、テープが使われるべきである情報を備えたソフトウェアと、テープの初めと関連付けられるべき絶対時間を提供するかもしれない。）タイムストリームオブジェクトへのインターフェースは、以下の通りである。

```

tream Sequence,
getSessionByName(name: Session),
getAllSessions() : Session
Sequence,
createEvent(start: Time, du
r: Time, type: String, props:
PropertyList) : Event
END
【0030】タイムストリームアーキテクチャは、プレーヤとレコーダオブジェクトがメディア型から独立して
いる共通のインターフェースを共有しており、また、サポートされたメディアの型のセットが固定されていない
という意味で拡張性がある。プレーヤとレコーダオブジェクトへの単純化されたインターフェースは、以下の通りである。

```

```

METHODS
seek (t: Time),      再生時間を t に変更する
location () : Time,   最新の再生時間に戻る
play (s: Speed),     速度係数 s で再生する (1=通常)
stop (),              再生を停止する
.
.
.
attach (timestream: Timestream) 所定のタ
イムストリームを再生するためにプレーヤをセットする
END;
TYPE Recorder=CLASS
METHODS
record (),
pause (),
createRecording (session: Session,
name: Name) : Timestream,
finishRecording (),
END;

```

【0031】所定のメディア型についてのプレーヤとレコーダオブジェクトは、そのメディア型と関連した' MediaServer' によって実施される。当然、新しいメディア型をサポートすることは、新しいMediaServerの実施を必要とするが、アーキテクチャ（すなわち、レコーダとプレーヤを作り出すか制御するためには必要なインターフェースへ）の変化は必要とされない。これは、汎用のアプリケーション（例えば、記録制御プログラム、或いは、アクセス及び再生制御プログラム）を書き込むことができ、新しいメディア型、或いは、存在するメディア型についての新しいサービスの実施を自動的にサポートすることになることを意味する。MediaServerオブジェクトへのインターフェースの単純化された記述は、下式により与えられる。

```

TYPE MediaServer= CLASS
METHODS
getRecorder (format: String, context: String) : Recorder,
getPlayer (format: String, context: String) : Player
END

```

【0032】文脈の引数は、プレーヤ或いはレコーダに関する情報を指定する。例えば、オーディオ入力／出力のために使われたホスト及び装置名を指定することができる。

【0033】アナライザは、一つ或いはそれ以上のタイムストリームと関連したデータを処理して、イベントを作り出すために使うことができる。現時点での好適な実施態様で使われた二つのそのようなアナライザが、以下に述べられる。アナライザへのインターフェースの単純化された記述は、下式で与えられる。

```

TYPE Analyzer=CLASS
METHODS
produceEvents (session: Session, specs: String)
END

```

【0034】アナライザは、所定のセッションのタイムストリームのデータにアクセスし、セッションと関連したイベントを生成する。どの型のタイムストリーム（例えば、オーディオ、ビデオなど）が、アナライザの型に依存してアナライザによりアクセスされる。' specs' の議論は、分析が行なわれる方法についてのあらゆる必要な情報を提供する。例えば、アナライザは、所定のワード（specsストリングの一部である）が話される度にイベントを生成するオーディオのワード監視器とすることができます。アナライザは、セッションが記録された後に、或いは、記録プロセスの間にリアルタイムで動作させることができることに注意すべきである。リアルタイムの場合には、アナライザによって作り出されたイベントは、ユーザインターフェースを実施するプロセスのような他のプロセス（以下に述べるイベント通知を参照のこと）により' 観察' することができる。

【0035】この拡張性を更に促進するために、アーキテクチャは、指定された型のメディアをサポートするための適切なMediaServersを配置するプローカオブジェクトを提供する。プローカは、同様にアナライザのような他の種類のオブジェクトを配置するのを補助することができる。プローカは、共有されたネットワークファイルに蓄積される情報を使って、実行プロセスが所定のオブジェクト型（例えばオーディオのMediaServer）を実施し、及び／又は、これらのプロセスを生成するために実行しなければならないトラックを維持することができる。他のネットワークベースのホームサービスも同様に使うことができた。プローカへの

インターフェースの単純化された記述は、下式で与えられる

```
Type Broker=CLASS
METHODS
getMediaServer (media: String) : MediaServer
getAnalyzer (type: String) : Analyzer
END
【0036】イベント、タイムストリーム、及び、セッションデータは、タイムストリームデータベースに蓄積される。タイムストリームデータベースを使うアプリケ
... 'マスター' と呼ばれるトップレベルタイムストリームオブジェ
クトを得る...
    session=master. getSessionByName ('R
ecordings')
    broker=master. getBroker ()
    mserver=broker. getMediaServer ('aud
io')
    recorder=mserver. getRecorder ('mu
aw', 'default audio device')
    recorder. createRecording (session,
'recording1')
    recorder. record ()
    ... あるコードは、記録がポーズ、レジューム、或いは、完了であ
るべきことを示すユーザインターフェースイベントを探す。
    ... 記録を閉じるためのルーチンをコールする。
    recording. finishRecording ()
    exit
```

【0038】前述の疑似コードフラグメントは、記録と呼ばれる現存するセッションの' recording 1' と名付けられたオーディオ記録を生成するために必要とされるステップを述べる。

【0039】記録を再生するために必要な疑似コードは、以下の通りである。

```
... マスターと呼ばれるトップレベルタイムストリームオブジェクトを得る。
session=master. getSession
ByName ('Recordings')
timestream=session. getTim
estreamByName ('recording
1')
broker=master. getBroker ()
mserver=broker. getMediaSe
rver ('audio')
player=mserver. getPlaye
r ('mu_law', 'default audio
device') player. play (1)
再生が停止されるべきこと、或いは、異なった時間の探
索がコールされるべきことを示すユーザインターフェース
```

ーションプログラムは、他のオブジェクトを作り出すか配置する方法を有するタイムストリームマスター オブジェクトを最初に得る。マスター オブジェクトは、上述したプローカによって使われたものと同じ機構を使用する（多分別のホストの上の）別のプロセスで実行されるタイムストリームサーバから得ることができる。または、タイムストリームサーバ（及び、マスター オブジェクト）を実施するプログラムコードは、アプリケーションプロセスに直接に含ませることができる。

【0037】疑似コードによる簡単な記録アプリケーションのために必要な基本的なステップは、以下の通りである。

... あるコードは、記録がポーズ、レジューム、或いは、完了であ

るべきことを示すユーザインターフェースイベントを探す。

... 記録を閉じるためのルーチンをコールする。

```
recording. finishRecording ()
exit
```

イベントを待つ。

【0040】タイムストリームアーキテクチャは、一般に互いに協調するアプリケーションを補助し、特に、多数のプレーヤとレコーダに対する協調制御を補助するいくつかの機構を提供する。1つの機構は、单一のプレーヤ及び/又はレコーダとして動作するプレーヤ及び/又はレコーダの収集体である連合（Confederation）オブジェクトである。プレーヤ或いはレコーダは、動的に連合に付加され、或いは、連合から削除することができる。連合へのインターフェースの単純化された記述は、下式により与えられる。

```
TYPE Confederation=CLASS
SUPERCLASSES Player, Reco
rder END
METHODS
registerPlayer (player: Pla
yer), unregisterPlayer (pla
yer: Player), registerRecor
der (recorder: Recorder), un
registerRecorder (recorde
r: Recorder), . . . END
```

【0041】プレーヤとレコーダのサブクラスとして、連合は、再生と記録のようなそれらのクラスの方法をサポートすることが暗示される。また連合は、時間範囲のシーケンスの再生として振る舞い (behavior) を実施する付加的な方法もサポートする。

マスターと呼ばれるトップレベルタイムストリームオブジェクト

```
... session=master. getSessionByName(
'ProjectMeeting#3')
... broker=master. getBroker()
... conf=master. getConfederation('xyz')
... timestreams=session. getAllStreams()
for each timestream in timestreams do
begin
... mserver=broker. getMediaServer(timestream. mediaType())
... player=mserver. getPlayer(timestream. format(), 'host1')
... conf. registerPlayer(player)
end
conf. play()...
```

どのように再生が制御されるかを示すユーザインターフェースイベントを待つ。

【0043】連合には'xyz'という名前が与えられることに注意すべきである。他のプログラムは、マスター オブジェクトからこの連合を得、これを使用して再生を制御することができる。これは、同じホストの上でいくつかのプログラムを実行することを可能とするが、しかし再生を共同して制御するために、多分、非常に異なったインターフェースを必要とする。なお、上述したのと同じであるが、異なるホストの上で実行されるプログラムは、そのホストと関連してプレーヤを生成することができ、それらを'xyz'構成と共に登録することができる。結果は、実施するための追加のプログラミング負担を本質的に必要としない共有分散型の再生ツールとなる。

【0044】別の整合機構は、イベント通知である。プログラムは、'noticeEvent'に呼ばれる方法を有するオブジェクトであるEventNotifierを実施することができる。ノティファイアは、セッションオブジェクトと共に登録することができ、ついで、イベントがそのセッションに付け加えられる度に、その時イベントで引数としてイベントと共にコールされる。例えば、LiveBoardウインドウにおいて、いくつかのオブジェクトが選択されるときに、選択されたオブジェクトに関連した時間を示すイベントを生成することができ、登録されたEventNotifierを備えたタイムラインインターフェースに接触すること

【0042】連合を使って、'ProjectMeeting#3'と呼ばれるセッションについて、記録された全ての媒体の再生をセットして制御するための単純化されたコードは、以下のようになる。

ができる（すなわち、コールされたnoticeEvent方法）、これらの時間をグラフィック的に示すことができる。この例が、以下に詳細に述べられる。

【0045】同期の問題は、種々の接合点で、また、種々の形態で発生する。最初に、キャプチャの間、エラーが再生の間知覚的に苛立たせていないように、タイムストリームを十分に整列した状態にしておくという必要性がある。オーディオとビデオの場合には、これは、ビデオと音が約1/30秒で（例えば、苛立たしい唇同期の問題が発生する前に）整列させ続けなければならないことを意味する。二つのタイムストリームは、異なるマシン上で独立して獲得される場合があるので、各種のソースの間で或る共通のクロックを使う（または再建する）ことが必要である。現時点での好適な実施態様においては、完全に接続されて実行しているときには、RFC-1769で述べられたネットワーク時間プロトコル（NTP）が、クロックをミリ秒以内に精密に維持するために分散型マシンの間で使用される（これは適応型のプロトコールであり、単に共通の標準に再同期させることによってではなく、マシンのドリフトにどの様に適合するかを学習することによって、通常より良い結果が得られる。好適には、タイムストリームのデータの各々の量はタイムスタンプされるが（ビデオではリアルタイムプロトコルパケットを使ったので）、データが正確なクロックレートでサンプルされるというシステムの確実性も同様に十分である。

【0046】他の場合には、例えば、他のメディアタイ

ムストリームにインデックスを付ける文の注釈の場合には、そのような厳密な同期のための要求を緩和することができる。再び、どのようなエラーも再生の間に苛立たせなることがない十分な正確さを必要とした測定基準が適用される。文のノートについては、これは、マークが記録された材料へ機能的なポインタであるように、ノートは、ユーザが意図したものに十分に近い位置で、タイムストリームにインデックスを付けなければならないことを意味する。或る状況の下では、何秒かのこぼれは許容可能であるが、他の場合には1秒の端数でも許容できない場合がある。もしユーザが注釈ソフトウェアを走らせてているハードウェアの上に、NTPを走らせていれば、その測定は、一般に十分な正確さを提供するだろう。他のオプションは、注釈がされる度に、NTP装備マシンから現在の時間を得るために、時間サーバをコールするか、または、注釈マシンとNTP時間との間のオフセットを計算するために、ミーティングの開始にクロックを同期させることを含む。この最後の形態の訂正是、特別な“同期イベント”的形態を使用するかもしれない。これは、分散型の資源の二つ或いはそれ以上のデータ内に注記されている、或いは、それから分析することができる。

【0047】要求される同期の正確さの程度に殆ど無関係に、種々のキャプチャ装置を完全に切り離した状態で動作させる準備ができる。これは、接続（コンピュータ、オーディオ、ビデオ等）が望ましくないか不可能なセッティングにおいて記録が行われる場合には望ましいことである。装置の異種のクロックを最終的に整列させるための手段が提供されるとすれば、ハンドヘルドコンピュータ或いは商用のカムコーダは、受け入れができるキャプチャ装置である。切り離されたラップトップをオーディオの記録と一緒に整列する技術の例は、騒々しくラップトップキーを打つことであり、有効な同期イベントである。キーストロークは、ラップトップによってピックアップされ、そのクロックによってタイムスタンプされ、オーディオは、キーヒットを表現するオーディオイベントを見つけ出すために分析され、このイベントは、オーディオレコーダのクロックによりタイムスタンプされる。これから、二つのクロックの間のオフセットを計算することができる。そして、タイムスタンプのセットの一方が他方に合わせられる。

【0048】ドリフト（非常に低い周波数エラー）は、記録するセッションの間、同様に問題である。独立したクロックが採用されたセッションにおいて必要とされる正確さの程度に依存して、システム的な利得或いは損失を補正するために、或いは、オフセット補正のためにクロックを繰り返しチェックするために、セッションの開始と終了を読み取ることが必要かもしれない。もし調整的な動作（例えば、通常速度再生が所望のときは、8057/8000で再生）を提案するためにデータを収

集することができれば、それら自身がクロック動作するシステム（例えば、8057サンプル/秒となるべきであるが8000サンプル/秒で動作しているとされたオーディオ装置）のこれらの要素におけるドリフトは、再生の間に付加的な微調整を必要とするかもしれない。

【0049】ジッター（一層高い周波数エラー）は、一般にシステムの適切な場所でバッファすることによって処理される。

【0050】タイムストリーム再生セッション（キャプチャセッションとしばしば同時に発生したり、または、キャプチャのセッティングにおいてある種の“入れ子状態”となるかもしれない）における同期の問題は、マルチメディアシステムにおいて典型的に取り扱われるものに一層接近している。

【0051】文脈をセットするために、各種のメディアタイムストリーム（オーディオ、ビデオ、計算ログ）が獲得され、獲得された活動が再体験できるように、密接に同期してこれらのタイムストリームの全部或いは所望のサブセットを再生しながら蓄積されると仮定する。更に、全て記録が完全なデータを備えており、補正すべきドリフト、ジッター、または、オフセットがない状態で獲得されたと仮定する。

【0052】現時点での好適な実施態様においては、種々のタイムストリームプレーヤは、適度に接近して同期させるために、十分に同期した無ドリフトシステムクロックを使用して、フリーホールモードで動作する。これは、各々のプレーヤが、システム時間の経過に合う方法で、そのタイムストリームを再生する責任があることを意味する。普通の再生スピードにおいては、これは再生（及び、システム時間経過）の1秒が、オリジナルディスク記録の各々の秒毎に正確に発生すべきであることを意味し、一層速い及び一層遅い再生レートは、再生される記録が一層多いか一層少ないことを単に表す。これらの高い及び低いスピード再生の最も良い表現は、ユーザの必要に応じてカスタマイズすることができることに注意すべきであり、高速のピッチ保存スピーチは、ミーティングをざっと知るために望まれる場合があり、或いは、適切な間隔から連鎖する4分の1秒のビデオクリップは、非常に高速のビデオ再生を最もよく伝えるかもしれない。

【0053】フリーホール同期を使うことの二つの局面は、記するに値するものあり、それは良好な開始と休止の機会である。第1のものは、伝送時間、バッファリング等に無関係に、そのようなアプローチの訂正動作は、全てが同時に開始するプレーヤに単に依存することである。これは、プレーヤが予定された開始についての依頼を受け取るべきであることを意味する（例えば、Yシステム時間で記録された材料の100%の通常速度でXシステム時間（少ないけれども未来への十分な時間（秒の端数）に再生を開始する）。開始のこのモード

は、次に続くリアルタイムクロック固守（Adherence）が所望の動作を生み出す調和した開始を保証する。休止の機会は、単に、特定のユーザ行動がプレーヤに分同期補正（例えば、マスターであるべき一つのプレーヤを宣言し、ユーザが再生を停止或いは休止する度に、全てのチャイルドプレーヤをそれに同期させる。

【0054】本発明のハードウェアーアーキテクチャのブロック図は、図1を参照して説明される。図1を参照すると、セッション始動モジュール101は、セッションを記録する準備として種々の“ハウスキーピング”機能を行なう。セッション始動モジュールは、好適には、種々の記録構成要素に連結されるコンピュータに基づいたシステム上で実行されるソフトウェアとして具体化される。そのタスクの間で、セッション始動モジュールは、セッション蓄積装置103が生成することができる一時的データを蓄積するための十分な利用することができる蓄積容量を有することを確かめなければならない。これは、記録されるセッションの長さ、及び、使用されるキャプチャ装置の数／型に関する推定を行うことにより達成される。キャプチャセッション始動モジュール101は、各々のキャプチャ装置102a-cによって“記録”の開始を同様に同期させる。各々のキャプチャ装置は、同時に、或いは、異なった時間に記録を開始することができる。

【0055】キャプチャ装置102a-cは、種々のメディアのタイムストリームを獲得し、或るインスタンスにおいてはイベントを生成する。キャプチャ装置102aは、タイムストリームとイベントデータ107を発生させる。現時点での好適な実施態様においては、キャプチャ装置102aの例は、カリフォルニア州、サンホセのゼロックスカンパニー（Xerox Company）のLiveWorks（商標）から入手可能な市販の製品LiveBoard（商標）のような電子ホワイトボードである。相互作用が発生するに従って、LiveBoardから獲得されたタイムストリームデータは、非同期的に発生する。イベントも、所定の機能（例えば、LiveBoardの上でページの変更）の呼び出しによって同様に発生させられる場合がある。

【0056】キャプチャ装置102bは、タイムストリーム108を発生させる。そのような型のキャプチャ装置においては、キャプチャ装置は絶えずタイムストリームデータを獲得するモードにある。キャプチャ装置102bの典型的な例は、ビデオ及びオーディオのレコーダである。そのようなキャプチャ装置102bの別の例は、ポインタである。ポインタキャプチャ装置は、ポインタ、例えば、投影されたスライド或いは他の視覚部材の上で何かを指し示すために、プレゼンテーションの間に典型的に使われるレーザーポインタの動きを獲得するものである。ポインタキャプチャ装置は、一般的に見られる材料が前もって用意されているときに、正式のプレ

ゼンテーションのために特に有益である。ポインタタイムストリームの再生は、このようにプレゼンテーションの再生の間のポインタの動きを示す。

【0057】キャプチャ装置102cは、イベント109を単に発生させる。キャプチャ装置102cの例は、ボタンである。ボタンは、ボタンタイムストリームのイベントの生成を、ボタンを操作するユーザと関連づける装置である。ボタンイベントは、興味があるトピックの開始、特に有益な対話、トピックの切り換わりのような種々の活動を指示するために、ユーザによって生成される場合がある。別の例は、スライドプロジェクトである。スライドプロジェクトタイムストリームにおけるイベントは、スライドの変更を指示する。

【0058】キャプチャ装置102a-bは、好適には、デジタルフォーマットでタイムストリームを獲得する。タイムストリームの種々の部分への直接的且つランダムなアクセスを容易にするため、デジタルフォーマットでタイムストリームを蓄積することは望ましい。しかしながら、アナログフォーマットでタイムストリームデータを獲得して蓄積することは、本発明の範囲内である。タイムストリームのフォーマットは、タイムストリームを生成するために使用されるキャプチャ装置に対応する。例えば、オーディオのタイムストリームは、或る規則的な間隔で取られた一組のオーディオのサンプルから成るが、一方、LiveBoardタイムストリームがタイムスタンプされたプリミティブな動作（以下よりの詳細に説明される）の履歴ファイルである。獲得されたタイムストリームデータの管理は、どのような方法であるにせよ、それらが最も適切であると見做す方法で、それらのデータを蓄積するために選択することができる種々のメディアサーバによって行なわれる。

【0059】生成された一時的データは、セッション蓄積装置103に蓄積される。セッション蓄積装置103は、データの異なった型、すなわち、セッションデータ、イベントデータ、及び、タイムストリームデータについて永久的な蓄積装置を提供する。セッション蓄積装置103は、実際には、異なった物理的な蓄積装置に分散され、タイムストリームアーキテクチャ（以下に述べる）の異なった構成要素によって管理される。

【0060】同様に、セッション蓄積装置103に連結されているのは、一時的データアナライザ104である。一時的データアナライザ104は、イベント110を識別するために、一時的データ111を分析するためには使われる処理手段（例えば、コンピュータに基づいたシステム）である。イベント情報は、セッション蓄積装置103に同様に蓄積される。

【0061】記録されたセッションへのアクセスは、セッションアクセスワークステーション105により達成される。このアクセスは、ネットワーク（例えばクライアント-サーバーアーキテクチャで）を介して、或いは、

直接接続とすることができる。アクセスセッション始動モジュール109は、セッションを再生するために必要な種々の“ハウスキーピング”機能を行う。アクセスセッション始動モジュールの機能性は、典型的にはセッションアクセスワークステーション105によって行なわれる。セッションアクセスワークステーション105は、典型的にはコンピュータ制御型ディスプレイシステムであり、そこでは、セッション再生に対する制御は、ディスプレイ上に表示されたグラフィックユーザインターフェースにより達成される。そのようなグラフィックユーザインターフェースは、以下に述べられる。現時点での好適な実施態様においては、セッションアクセスワークステーションは、カリフォルニア州、マウンテンビューのSUN Microsystems Inc. から入手可能なX-Windowsグラフィックデザイナーフェースを備えたSun OS 4.1オペレーティングシステムが実行されるSUN SparcStation-10である。

【0062】更に、セッションアクセスワークステーション105に連結されているのは、プレーヤ106、再生制御装置107、及び、エディタ108である。種々のプレーヤ、再生制御装置、及びエディタの各々は、セッションアクセスワークステーションの中で統合される。そこで、電子ホワイトボードタイムストリームの再生は、ワークステーションのディスプレイ上の第1のウィンドウにより達成され、ビデオタイムストリームはワークステーションのディスプレイ上の第2のウィンドウにより達成され、オーディオタイムストリームは、ワークステーションのオーディオサブシステムにより達成される。しかしながら、プレーヤはセッションアクセスワークステーションの中で統合される必要がなく、個別の再生装置とすることは当業者にとって明白だろう。更に、現時点での好適な実施態様においては、単一のウィンドウが、プレーヤ、再生制御装置、及び、エディタの機能を行なうことができる。現時点での好適な実施態様におけるウィンドウの組織及びそれらに付随する機能性は、図3を参照して以下に詳細に述べられる。

【0063】最後に、エディタの使用により生成された新しい一時的データ112は、セッション蓄積装置103に蓄積される。

【0064】共同作業におけるLiveBoardの主要な機能は、各々のユーザが見ることができて、相互に作用することができる“共有表現”を提供することである。LiveBoardとそのオペレーティングソフトウェアは、そのような“共有表現”が単一のLiveBoard装置の上にある必要がないように設計されることに注意すべきである。実際の共同作業が異なる物理的な位置で行われるように、多数のLiveBoardsがネットワーク経由で一緒に連結することができ

る。

【0065】現時点での好適な実施態様において使われたようなLiveBoardは、主にセッション記録へインデックスを生成するための手段であるようには意図されていない。しかしながら、以下に一層詳細に議論されるように、LiveBoardは、それを記録されたセッションの次の再生を制御するための有益なツールにする独特の能力を有する。LiveBoardは、Tivoliアプリケーションの制御の下で動作する。Tivoliアプリケーションは、Pedersen、E.、K. McCall、T. Moran、及び、F. Halaszによる“Tivoli: An Electronic Whiteboard for Informal Workgroup Meetings”, Proceedings of the INTERCHI'93 Conference on Human Factors in Computing Systems, pp. 391-389, April 1993に詳細に説明されている。

【0066】Tivoliは、“描画”プログラムと知られている画像生成プログラムのクラスに似ているユーザインターフェースを有している。描画プログラムは、グラフィックオブジェクトの収集物のような画像を生成し操作する。Tivoli画面は、グラフィックオブジェクトとして描かれる。グラフィックオブジェクトは、以下のものを含む。

—ストローク：ペン入力装置により生成された“デジタルinking”オブジェクト。

—文字：キーボード経由で生成されるか、テキストファイルから持ち込まれた。

—画像／アイコンオブジェクト：持ち込まれた或いはコピーされた。

—構造化オブジェクト：データベースから持ち込まれた。

—クロック：クロックジェスチャーと共に生成される。クロックは、その生成時間を表す丸いアナログ時計文字盤として現れるLiveBoardの上で生成される特別なマークである。

【0067】Tivoliは、実行されたアリミティブな動作の履歴リストを維持する。例えば、オブジェクトを付け加える、オブジェクトを削除する、オブジェクトを生成する、オブジェクトを変える、オブジェクトを選択する、または、ページを変える。履歴リストの全てのアリミティブな動作は、タイムスタンプされてオブジェクトと関連付けられる。以下に詳細に述べられるように、イベントを識別するために分析されるのがこの履歴リストである。

【0068】LiveBoardに現れるグラフィックオブジェクトは、ミーティングのコースの間に作られるか、或いは、パーソナルコンピュータ上のワードプロセ

ッキングファイルを開ける操作に似た方法で、或るメモリ媒体（例えば、フロッピーディスク）からロードされるかのいずれかとすることができます。この後者のインスタンスは、セッションが正式のプレゼンテーションであるときに、特に有益である。

【0069】LiveBoardの動作の別の重要な局面は、ジェスチャーの使用である。ジェスチャーは、例えば他の任意の“インクストローク（inkstroke）”のように生成されるが、コマンド（例えば編集コマンド）として解釈されるLiveBoardとの相互作用である。例えば、もし特定のオブジェクトが“削除”されるべきであれば、オブジェクトはペンにより囲むジェスチャーと、呼び出された削除コマンド（例えば、ピグテールジェスチャー）で選択される。削除が生じた後、ジェスチャーによって引き起こされたマーキングは、もはやLiveBoardには現れないしかしながら、後で述べられるように、本発明においては、セッションが再生されるとき、“ゴースト”画像が残る場合がある）。

【0070】上述したように、セッション蓄積装置は、異なった型のデータ、すなわち、セッションデータ、イベントデータ、及び、タイムストリームデータについての永久的な蓄積を提供するタイムストリームアーキテクチャの機能構成要素である。セッション蓄積装置は、実際には、異なった物理的な蓄積装置に分散され、タイムストリームアーキテクチャの異なった構成要素によって管理される。セッション蓄積装置を、タイムストリームデータベース（セッションとイベントデータ用）、及び、バルクタイムストリーム蓄積装置（タイムストリームデータ用）に分けることが最も実際的である。

【0071】バルクタイムストリーム蓄積装置は、タイムストリームの実際のデータが蓄積され管理されるところである。ビデオの場合には、これは、デジタル化されて圧縮されデジタル化されたビデオフレームの長いかさばったシーケンスから成る。タイムストリームアーキテクチャにおいては、タイムストリームデータの管理は、どのような方法であれ、それらが最も適切であると見做す方法でそれらのデータを蓄積するために選択することができる種々のメディアサーバに任される。現時点での好適な実施態様においては、大部分のメディアサーバは、それらのデータを通常のUnixファイルに蓄積するが、他の蓄積機構（例えばオーディオ/ビデオタイムストリームのアナログ蓄積のためのコンピュータ制御型VCR）のためのメディアサーバも同様に実施された。メディアサーバについては、断続した少容量データ（例えば、ラップトップコンピュータによって作り出された文のノート）を備えたタイムストリームのための最も便宜的な機構とすることができます。そのタイムストリームデータのためのタイムストリームデータベースを使うことも可能である。

【0072】タイムストリームデータベースは、セッションとイベントデータ、すなわち、全ての情報をセッションに組織化し、セッションの要素を記述するデータ、を蓄積して検索するための蓄積場所である。タイムストリームデータベースは、ユーザが、所望のセッションを位置決めし、どのタイムストリームがそのセッションに関連付けられているのかを決定し、セッションとそのタイムストリームに連付けられているイベントを見つけることをサポートする。タイムストリームデータベースは、セッションデータの閲覧をサポートするだけでなく、セッションとイベントの選択的な検索を可能と質問能力を有するデータベースであるべきである。セッションとイベント情報は、数ヶ月、数年にわたって持続しなければならず、したがって、不揮発性装置（通常はディスク）に蓄積されなければならない。タイムストリームアーキテクチャはオブジェクト志向であるので、オブジェクト志向のデータベースが、タイムストリームデータベースのための持続性があるオブジェクト蓄積装置として適切である。現時点での好適な実施態様においては、タイムストリームデータベースは、Unixファイルシステム上で実施される。

【0073】一時的データ分析は、抽出すべきタイムストリームと所望のイベントによって獲得された情報に対して特定のものである。オーディオの情報のタイムストリームの場合には、イベントの一つの有益な形態は、特定の人がいつ話しているかを知ることである。従って、1) 異なった人がいつ話しているかを表すためにタイムストリームをセグメント化し、2) 話している人に種々のセグメントを割り当てて、オーディオストリームを分析することが望ましい。したがって、これらのセグメントは、タイムストリームにおけるイベントを表現することになる。オーディオストリーム上でそのような分析を行なうための技術は、両方とも本出願と同じ譲受人に譲渡されている“Unsupervised Speaker Clustering For Automatic Speaker Indexing Of Recorded Audio Data”と題された、同時競続中の出願番号8/226, 525号、及び、

“Segmentation Of Audio Data For Indexing Of Conversational Speech For Real-Time Or Postprocessing Applications”と題された出願番号8/226, 519号に説明されている。いずれの場合でも、いったんそのような分析が行なわれると、オーディオのタイムストリームのためのイベントのリストは、それがタイムスタンプ、期間、及び、話者識別子を含む複数のデータ項目から成ることになる。

【0074】LiveBoardによって獲得されたタイムストリームの分析の場合には、イベントは、図面の

描画或いは言葉の書き込み、ページ変更、図面或いは言葉の消去、既存のマーキングに対してなされた注記、或いは、特定のコマンドのLiveBoardユーザインターフェースへの入力とすることができます。上述したように、LiveBoardは、Tivo1iアプリケーションの制御の下で動作する。Tivo1iによって発生させられたタイムストリームは、一時的な順序で記録されたタイムスタンプされたプリミティブな動作の履歴リストである。そのようなタイムストリームの分析は、意味のあるイベントを識別するための、プリミティブな動作のシーケンスを構文解析することを伴う。

【0075】LiveBoard上の各々のマーキングは、オブジェクトと呼ばれる。各々のオブジェクトは、それと関連づけられた一つ或いはそれ以上のイベントを有する。本発明にとって、オブジェクトと関連している全てのイベントが識別され、セッションの中へのインデックスとしてユーザーに提供されることが必要である。セッションの中へのインデックスとしてユーザーに確認され、提供されることが、必要である。従って、本発明における一時的データ分析は、オブジェクト上の種々のイベントを識別し、そのイベントを蓄積する。全てのオブジェクトについて、オブジェクト生成イベントが存在する。オブジェクトは、オブジェクト変更イベントとオブジェクト削除イベントを同様に有することができる。LiveBoardタイムストリームの更なる局面は、LiveBoardウィンドウに関して以下に議論される。

【0076】現時点での好適な実施態様においては、セッションアクセスワークステーションは、蓄積媒体上に蓄積されるプログラムインストラクションの制御の下で動作するコンピュータ制御型ディスプレイシステムであり、図2を参照して述べられる。図2を参照すると、コンピュータに基づいたシステムは、バス201経由で連結される複数の構成要素から成る。ここで説明されるバス201は、本発明を不明瞭にしないために、単純化される。バス201は、(例えば、プロセッサバス、ローカルバス、及び、I/Oバス)の階層構造と同様に、複数のパラレルバス(例えば、アドレスバス、データバス、及び、ステータスバス)から構成することができる。いずれの場合でも、コンピュータシステムは、内部メモリー203(内部メモリー203は、典型的には、ランダムアクセスメモリー或いはカーリードオンリーメモリーの組み合わせであることに注意すべきである)からバス201経由で提供されるインストラクションを実行するためのプロセッサー202を更に含んでいる。動作中に、本発明の種々の機能構成要素を実行するためのプログラムインストラクションは、内部メモリー203に蓄積される。プロセッサー202と内部メモリー203は、個別の構成要素、或いは、用途特定集積回路(ASIC)チップのような単一の集積装置とすることができます

る。プロセッサー202と内部メモリー203は、ここに述べられる種々の処理機能を行なうための電気回路から成る。

【0077】同様に、バス201へ、英数字を入力するためのキーボード204、データを蓄積するための外部蓄積装置205、カーソルを操作するためのカーソル制御装置206、及び、視覚的出力を表示するためのディスプレイ207が連結される。キーボード204は、典型的には標準のQWERTYキーボードであるが、同様に、電話機のようなキーパッドにすることもできる。外部蓄積装置205は、固定の或いはリムーバブルの磁気或いは光学ディスク装置とすることができる。カーソル制御装置206は、典型的には、或る機能の実行をプログラムすることができるそれと関連したボタン或いはスイッチを有する。更に、バス201へ、オーディオ出力手段208が連結される。オーディオ出力手段208は、典型的には、何らかのオーディオ信号発生手段(例えば、デジタル・アナログ変換器)及びスピーカーから成る。オーディオ出力手段208は、プレーヤとしても動作することができる。最後に、バス201に連結されるのが、ビデオ出力手段209である。ビデオ出力手段は、典型的には、ビデオ信号をディスプレイ207に表示できるようにするための市販の処理手段、例えば、ビデオボードである。

【0078】セッションアクセスワークステーションが、蓄積媒体に蓄積されたプログラムされたインストラクションに従って動作するコンピュータに基づいたシステム上で具体化されているが、本発明は、固定機能のターミナル或いはLiveBoardのような任意のコンピュータ制御型ディスプレイシステム上で実行することができる。

【0079】セッションアクセスワークステーションは、上述したように、タイムストリームアーキテクチャを参照し、再生と制御のためのプログラムインストラクションを使って動作する。図3は、セッションアクセスワークステーションの基本的なローレベル動作を説明する単純化された状態図である。図3を参照すると、セッションアクセスワークステーションは、セッションを再生する依頼が受信されるとき、最初は、開始状態301にある。次いで、セッション始動状態302に入る。セッション始動状態302の間、再生されるべきセッションがユーザーによって選択される。選択されたセッションの一時的データは検索され、種々のプレーヤ、制御装置、及び、エディタが同期化される。種々のプレーヤは、セッションアクセスワークステーションとして同じ物理的な装置に統合することができる、或いは、それらを分離した装置とすることもできることに注意すべきである。次いで、種々のウィンドウ(すなわち、プレーヤ、再生制御装置、及び、エディタ)は、セッションアクセスワークステーションに連結されたディスプレイに

表示される。この時点において、“作動”状態303と名付けられた状態に入る。ここで述べられた作動状態303は、セッションアクセスマネージメントが相互作用を行っており、セッションアクセスマネージメントについてのユーザインタフェースの種々の機能を何時でも呼び出すことができるという意図を伝えるために使われる。いずれの場合でも、最初に“作動”状態303に入ったときに、セッションには、初めからセッションを再生するためのキーが与えられる。“作動”状態303にある間は、ユーザ入力に応じて種々のプリミティブな機能が行なわれる。“再生”コマンド、または、等価なグラフィックユーザインタフェース入力（例えば特定のアイコンの上でクリックすること）の呼び出しあは、セッションの開始を引き起こすことになる（ボックス304）。これにより、再生状態は“再生”にセットされる。“停止”コマンドの呼び出しあは、セッションの再生を停止させ、再生状態は“停止”にセットされる（ボックス305）。特定のウィンドウにおける“ディスプレイ修正”コマンド或いは制御の呼び出しあは、ウィンドウを修正させる（ボックス306）。そのような修正は、タイムラインインターフェースで表示されたトラックを変えることができる。再生制御の呼び出しあは、再生点を変更させる（ボックス307）。これは、新しい点でキーを与えてセッションの再生を引き起こす。編集コマンドの呼び出しあは、既存の一時的データへの修正、或いは、新しい一時的データの生成のいずれかを引き起こす（ボックス308）。種々の状態における動作の開始により、作動状態303への復帰が生じる。

【0080】そのような多数の動作は、セッションアクセスマネージメント上で特定の機能を達成するためには、呼び出される場合があることに注意すべきである。例えば、LiveBoardウィンドウのオブジェクトを選択する動作は、“停止”に続いて“ディスプレイを修正”を呼び出すことができる。

【0081】最後に、動作状態303にある間、“退去”コマンドの呼び出しあは、アクセスセッションを停止させることになる（停止状態309）。

【0082】図4は、セッションアクセスマネージメントのスクリーンディスプレイを説明する。図4のスクリーンディスプレイは、同時に開いている複数のウィンドウを示す。ほとんどのグラフィックユーザインターフェース指向のウィンドウに関しては、ウィンドウのサイズと配置はユーザの選択事項である。図4を参照すると、同時に表示されているのは、LiveBoardウィンドウ401、ビデオウィンドウ402、タイムラインインターフェースウィンドウ403、ノートウィンドウ404-405、オーディオのウィンドウ406、マルチメディア編集ウィンドウ407、及び、ミーティングプレーヤウィンドウ408である。LiveBoardウィンドウ401は、プレーヤ、再生制御装置とし

て、或いは、エディタとして動作することができる。プレーヤとして、それはLiveBoardタイムストリームを再生するために使用される。制御装置として、それは、そこに表示されたオブジェクトとの相互作用を通してセッションの再生に対する制御を可能にする（以下に詳細に述べる）。エディタとして、ユーザは、新しいオブジェクトを生成するか、或いは、セッションの次の再生で表示されるセッションに関する特定の注釈を作ることができる。

【0083】ビデオウィンドウ402は、ビデオのタイムストリームを再生するために使用されるプレーヤである。ノートウィンドウ404と405は、ミーティングのコースの間のキャプチャ装置として使用可能なラップトップコンピュータ或いは同様な装置とすることができるノートを再生するために使用されるプレーヤである。

【0084】ミーティングプレーヤウィンドウ408は、プレーヤと再生制御装置である。ミーティングプレーヤウィンドウ408は、種々な他の一時的データのイベントを使って、それが再生されるにつれてミーティングのダイナミックスのシミュレーションを提供する。再生は、ミーティングプレーヤウィンドウ408に現れる種々のアイコンとの相互作用により制御される。ミーティングプレーヤウィンドウは、以下に詳細に述べられる。

【0085】オーディオウィンドウ406は、オーディオのタイムストリームの再生、特にセッションのオーディオ再生のための話者の音量を制御するために使用される制御装置である。

【0086】マルチメディアエディタ407は、タイムストリームとしてセッションに導入して戻すことができるノート、イベント、或いは、他の情報を生成するために使用されるエディタである。更に、マルチメディアエディタ407は、見ためのセッションアクセスマネージメントを必要としないマルチメディア文書を生成するため使うことができる。

【0087】タイムラインインターフェースウィンドウ403は、再生制御装置である。タイムラインインターフェースウィンドウ403は、イベントの一時的なものに基づいたビューを、それらがセッションの間に発生するに従って具体的に表現する。タイムラインインターフェースウィンドウ403は、以下に詳細に述べられる。

【0088】本発明の現時点で好適な実施態様のタイムラインインターフェースは、タイムラインに沿ったイベントの表現との相互作用により再生制御を提供する。イベントは、固定期間、或いは、期間不定（生成時間がだけが重要である）のいずれでもよい。セッション始動の間、イベントデータはセッションアクセスマネージメントの中へロードされる。イベントデータは、タイムライン上で縮尺調整されたビジュアルインジケータを生成するために使われる。これらのビジュアルインジケ

タは、再生をセッションの対応する時間に直接移動させるインデックス点として使用することができる。イベントデータをタイムライン表現に組織するための内部構造がここで述べられる。

【0089】ロードされたイベントデータは、ユーザにイベントデータを簡単な方法で制御させるために組織される。この組織は、型マップ、イベントメニュー、及び、トラックマップにより定義される。全てのイベント型は、*Types*と呼ばれる階層構造の名前を有する。例えば：

Speaker/Adam  
LiveBoard/Edit/Move  
Button/Betty  
Note/Laptop/Charlie

ユーザが定義した型マップは、そのような*Types*を生成するために使われる。型マップは、各々のイベントのパラメータを使用し、*Type*を作り出すルールのセットである。次いで、*Types*のセットは、時間トラックエリアのラベルエリア内でユーザに対して示される簡単な階層構造のイベントメニューの中へ整えられる。全てのイベントは、トラック上でユーザに表示することができる。レイアウトは、トラックマップにより定義される。トラックマップは、各々が対応するトラック上に表現される*Types*のリストを持つ複数のトラックのリストである。

【0090】ビジュアルインジケータは、各々の*Types*について生成される。ビジュアルインジケータは、どのようなグラフィックスタイルでもよいが、ビジュアルインジケータの二つの概略のクラスが最も一般的に使われる。第1のクラスは、セグメントと称される。セグメントは、イベントが生じる期間に対応するために、タイムラインの上でサイズ調整され、縮尺調整された長方形のボックスとして表示される。第2のクラスは、開始点と称される。これらは、左端がイベントが始まった時間の点に垂直に位置し、右端がイベントに対応する活動が無期限に続くことを表すために、右を“指す”三角形として表示される。そのようなイベントについては、イベント（例えば、人がノートを取りはじめるなどを示すイベント）の期間についての関心はない。種々の動作は、ビジュアルインジケータに関して行うことができる。ビジュアルインジケータの動作は、以下のものを含む。

- ビジュアルインジケータを生成する（所定の型又はデフォルトによる型）。
- ビジュアルインジケータを選択する。
- ビジュアルインジケータを選択から外す。
- 異なるトラックへビジュアルインジケータを動かす（時間は同じまま）。
- 異なるトラックへビジュアルインジケータをコピーする（時間は同じまま）。

—ビジュアルインジケータにより表現されたイベントに関する一層多くの情報を見る。

—選択されたビジュアルインジケータのディスプレイパラメータを編集する。

—ビジュアルインジケータ時間を編集する。

—ビジュアルインジケータ開始時間を調整する。

—ビジュアルインジケータ（セグメント）終了時間を調整する。

—ビジュアルインジケータ時間位置を調整する。

【0091】図5は、タイムラインウィンドウ内の基本的なエリアを説明する。図5を参照すると、タイムラインウィンドウ501は、複数のディスプレイエリアから成る。概観時間トラックエリア502は、セッションの概観を提供するために使われ、焦点時間トラックエリア503は、全体のセッションより少ないか等しい期間にわたって一層詳細な情報を提供する。各々のトラックエリアの動作が、同じ型の一時的な情報を提供することができることは注意すべきである。違いは、情報が意図する詳細さ（すなわち、全体のセッション対セッションの一部）のスケールである。

【0092】概観トラックエリア502内に定義されたのはクロックエリア512であり、焦点時間トラック503内に定義されたのはクロックエリア513である。対応する時間トラックエリアの時間スパンとスケールを示すために、各々のクロックエリア512と513は、クロック時間と印マークを現す。概観時間トラックエリア502は、その中に、焦点トラック508と称される特別なトラックを定義した。焦点トラック508の上にあるのは、焦点セグメント509である。焦点セグメント509は、焦点時間トラックエリア503に詳述されるセッションの期間に対応する。更に、それぞれのトラックエリアに関連するのは、ラベルエリアである。各々のラベルエリアは、対応する時間トラックエリアで表示される“トラック”を識別するためにものである。ラベルエリア504は概観時間トラックエリア502に対応し、ラベルエリア505は焦点時間トラックエリア503に対応する。概観時間トラックエリア502と焦点時間トラックエリア503の間の関係は、焦点エリア507で示される。この関係は、焦点セグメント509の終わりから焦点時間トラックエリア503の終わりへ広がるアーム510と511により、明白に目で見えるようになる。

【0093】同様に、図5で説明したのはボタンエリア506である。ボタンエリア506は、種々の再生制御機能を呼び出すための“ソフト”ボタンを含む静的なエリアである。そのようなボタンは、カーソル制御装置を使うことにより、ポイントしてクリックする方法で呼び出される。ボタンの配置と機能は、例示的なものであり、特定の一般に使用される機能への簡単なアクセスを提供することを意図している。

【0094】ボタン520は、再生を10秒スキップして戻るために使用され、ボタン521は、再生を10秒スキップして先送りさせる。ボタングループ522の方向を示す矢は、文脈に応じた移動或いはスクロールのために用意される。ボタン523は、セッションの再生を停止するために用意される。ボタン524は、セッションの再生を始める。ボタン525は、選択されたトラック或いは選択されたトラックのセットの上の次のイベントへスキップすることを可能とする。ボタン526は、編集モードに入って、タイムラインインターフェース、或いは、その上に表示されたビジュアルインジケータを修正するために用意される。ボタングループ527は、文脈に応じた削除或いはコピー（例えば、ビジュアルインジケータをコピー或いは削除する）のために用意される。ボタングループ528は、文脈に応じた拡張或いは折り畳み（例えば、トラックのセットを单一のトラックの中へ折り畳むか、または、折り畳まれたトラックのセットを拡大する）のために用意される。ボタン529は、セッションメニューを表示するために使われる。セッションメニューは、ユーザが、セッションのためにデータをロードし、セッションをセーブし、或いは、セッションを止めることを可能にする。

【0095】上記されたように、焦点時間トラックディスプレイエリアに示されるセッションの一部は、概観時間トラックエリアの焦点トラックの上に含まれる焦点セグメントにより示される。焦点時間トラックエリアにおいて焦点のスパンを変えるために、焦点セグメントはユーザによりサイズが修正される。これは、図6-7で説明される。図6において、焦点セグメント601は、焦点トラックエリア602に示されるセッションの一部を示す。この関係を識別する他のビジュアルキーは、それぞれのクロックエリア603と604に対する変化、すなわち、時間トラックエリアについての印マークとそれとの開始及び終了時間のサイズ修正、及び、時間インジケータ605と606の相対位置決めを含む。同様に、図6に示されるには、現在の再生時間を示す再生時間インジケータ607である。

【0096】焦点セグメントのサイズ修正は、種々のカーソル制御動作により達成される。現時点で好適な実施態様においては、カーソルが焦点セグメントの右端の上に位置している間にマウスのボタンが押されると、焦点セグメントはカーソルの移動の方向に引き伸ばされる。焦点セグメントの左側も、同じ方法で操作される。

【0097】図7は、焦点時間トラックエリアに示されたスパンを増加するために、右に引き伸ばされた焦点セグメント701の右側の結果の図である。この増加したスパンは、クロックエリア702と703の変化よりも同様に反映される（印マークの縮尺修正と焦点時間トラックエリアについての終了時間の変化）。

【0098】タイムラインインターフェースは、多数の

セッションの再生を制御するために、或いは、单一のセッションの多数のスパンに焦点を合わせるために使用することができる。図8は、二つのセッションのためのタイムラインインターフェースの図である。二つのセッションは、一日中のミーティングの午前半分と午後半分を表現するかもしれない。または、それらは、毎週のプロジェクト検討ミーティングのような定期的に予定されたミーティングの異なった出来事を表現するかもしれない。典型的には、二つのセッションは或る共通性を有する。図8を参照すると、タイムラインインターフェースウインドウ801は、二つの概観時間トラックエリア802と803と、対応する焦点時間トラックエリア804と805から成る。焦点セグメント806は、概観時間トラックエリア802に対応するセッションにおけるスパンと、焦点時間トラックエリア804におけるそのスパンについての詳細を示す。同様に、焦点セグメント807は、概観時間トラックエリア803に対応するセッションにおけるスパンと、焦点時間トラックエリア805におけるそのスパンについての詳細を示す。実際のトラックは、それぞれの焦点時間トラックエリア804と805で同一である必要はないことを示す。もしさらが異なっていれば、分離ラベルエリアが必要とされる。ディスプレイエリアで非常に多くのスペースを使用するので、これは望ましくない。

【0099】図9は、单一のセッションの多数のスパンの詳細が焦点時間トラックエリアに表示される場合があることを表す更なる図である。これは、多数の焦点セグメントを概観時間トラックエリアの焦点トラックの上に表示させる。図9を参照すると、第1の焦点セグメント901は、焦点時間トラックエリア内の対応するエリア903に示された詳細を有する一方、第2の焦点セグメント902は、焦点時間トラックエリア内の対応するエリア904を有する。そのようなディスプレイは、再生する興味がない或るエリアが存在することが判っているときには望ましい（例えば、LiveBoard上の特定のページに関する議論を検討したいだけということが前もって判っている場合）。デフォルトにより、焦点時間トラックエリアは、同じスケールの上で二つの焦点セグメントに均等に分けられる。しかしながら、図10で説明したように、セッション期間に割り当てられた焦点時間トラックディスプレイの一部を修正することができる。ここでは、図9のエリア904は、図10のエリア1001へ増加され、図9のエリア903は図10のエリア1002へ減少させられた。サイズのこの増加は、エリア904の左側を左に移動するドラッグ動作により行うことができる。

【0100】図11-図14は、以下のキャプチャ装置、すなわち、Liveboard、オーディオレコーダ、パーソナルコンピュータによる二つのユーザーテーリングノート、ビデオレコーダ、及び、各ユーザ毎のボタ

ン、から成るシステムについてのタイムラインインターフェースディスプレイ上で変化を示す。勿論、キャプチャ装置の他の組み合わせが利用される場合があり、また、幾つかのキャプチャ装置からのイベントは省略され、これにより、異なったタイムラインインターフェースディスプレイとなる。この例についてのセッション参加者は、アダム、ベティ、及び、チャーリーと名付けられる。

【0101】図11を参照すると、概観時間トラックエリアの1101は、全体のセッションについての情報を表示し、全体のセッションを示す焦点セグメントを有する。このようにして、焦点時間トラックエリア1102は、全体のセッションの詳細を表示する。再生インジケータ1113と1114は、それらのそれぞれの期間の中の現在の再生位置を示す。

【0102】更に、セッションエリア1101と関連があるのが、セッション識別子1103である。セッション識別子は、再生されるセッションを識別する。

【0103】概観セッション時間トラックエリアと関連している情報が提供され、従って、全体のセッションは、セッション開始時間1104とセッション終了時間1105を含む。セッション開始時間1104とセッション終了時間1105の各々は、セッションが起きた日の実際の時間を示すことができ、或いは、0:0:0に対する相対時間である(図11で説明したように)。

【0104】焦点時間トラックエリア1102と関連している情報は、スパン開始時間1106とスパン終了時間1107を含む。スパン開始時間1134とスパン終了時間1135の各々は、特定のスパンの日の実際の時間によるか、または、セッション開始時間0に対するものである。

【0105】焦点時間トラックエリア1102は、各トラックが何を表すかの情報を提供するためのトラックラベルセクション1111を含む。トラック1117と1118は、LiveBoard上で行なわれた編集イベントを表す。トラック1118は、LiveBoard上で行なわれる選択動作に応じて発生するものであり、以下に詳細に述べられる。トラック1117は、LiveBoard上のオブジェクトで行なった種々の編集動作を表す。トラック1117において、各々の編集動作は特にカラーコード化されたビジュアルインジケータを有する。そこで、赤色はオブジェクトの削除を表し、緑色はオブジェクトの付加を表し、青色はオブジェクトの移動を表す。

【0106】トラック1119-1121は、オーディオのイベントを示す。トラック1119は、“アダム”が話しているセグメントを示し、トラック1120は“ベティ”が話しているセグメントを示し、トラック1121は“チャーリー”が話しているセグメントを示す。参加者が話しているとき、セグメントの各々のビジ

ュアルインジケータは、視覚的に区別される(典型的には、色かテクスチャにより)。

【0107】トラック1122-1123は、ミーティングの参加者であるアダムとチャーリーにより取られたノートの開始点を示す。ノートは、キャプチャ装置として動作しているパーソナルコンピュータ、または、他のテキスト生成装置で取られた。

【0108】トラック1124は、“ボタン”押圧のイベントのインスタンスを示す。各々のミーティング参加者は、参加者が何か重要なものが議論されていると感じるときに押すことができるボタンを持っている。再び、ビジュアルインジケータは、ボタンを押圧した人を識別するために、カラーコード化される。そこで、例えば、青はベティが彼女のボタンを押圧したことを表し、緑はチャーリーが彼のボタンを押圧したことを表す。

【0109】トラック1125は、ビデオタイムストリームに関するイベントのインスタンスを示す。そのようなイベントは、人が話すために立ち上がる、或いは、ホワイトボードに関しての何らかの型の身振りをすることを含むことができる。そのようなイベントは、ビデオタイムストリームの分析により識別される。

【0110】上述したように、選択トラック1118は、LiveBoardウィンドウ内の選択相互作用の結果として自動的に発生させられる。選択相互作用は、LiveBoard上に表示される種々のオブジェクト(例えば、言葉或いは図)が選択されるところである。次いで、選択されたオブジェクトに関するイベントは、タイムラインウィンドウ上に表示される。同様に、選択が空間的エリアに関して生じるかもしれないことは、注意すべきである。そのような選択においては、エリアが選択され、選択されたエリアに配置されたオブジェクトについてのイベントは、タイムラインの上に表示される。これは、特にマーキングがLiveBoardから消されたセッションの一部の再生するために有用である(例えば、再生のプロセスの間、スキップが行われ、オブジェクトが消されたことに気付くか、または、それが別のオブジェクトと取り替えられた。)。図11に戻って参照すると、種々の選択イベントが説明される。再び、好適には、ビジュアルインジケータは、イベントの型を示すためにカラーコード化される。生成イベント1127は、オブジェクトが生成された時間の点を示し、移動イベント1128は、オブジェクトがLiveBoard上で動かされた時間の点を示し、カラー変化イベント1129は、オブジェクトカラーが変えられたことを示し、削除イベント1130は、オブジェクトが削除されたことを示す。

【0111】各々の時間トラックエリアの内容は、ユーザーが修正可能である。トラックは、併合し、付加し、隠蔽し、或いは、移動することができる。時間トラックエリアの内容を修正するために、動作は対応するラベルセ

クションの中で行なわれる。例えば、図11を参照すると、オーディオトラック1119-1121は、選択され、併合される場合がある。そのような併合されたトラックの結果は、図12で説明され、そこでは併合されたトラック1201が生成されている。トラックの選択は、カーソル制御装置を操作して、カーソル制御装置に関連したスイッチをスイッチを押す、カーソルをトラックのラベルを通過するように移動させることにより、達成することができる。ラベルの選択の指示は、逆のビデオモードで選択されたラベルを表すような、或る視覚的な刺激により達成される。選択されたラベルの識別は、スイッチを離すことにより達成される。この時点で、種々のトラック動作オプションが表示される。次いで、ユーザは所望の動作（この場合は、併合動作）を選択する。しかし、これはトラックディスプレイ動作を実行することができる一つの方法である。他の技術は、パルダウンメニューにより、或いは、コマンドラインから入力されたコマンドにより呼び出すことができる。

一トラックの動作は、以下のものを含む

- 一トラックを選択する
- 一トラックを選択から外す
- 一トラックを生成する（所定のH types）
- 一選択されたトラックを再配置する
- 一選択されたトラックを削除する
- 一選択されたトラックを（一つのトラックの中へ）折り畳む
- 一選択されたトラックを（そのH typesに基づいてトラックのシーケンスの中へ）拡張する
- 一選択されたトラックのディスプレイパラメータを編集する

一トラックラベルを編集する

【0112】異なるビジュアルインジケータの上のポイントとクリック動作は、異なる事を生じさせる。各々の場合において、タイムストリームの次の再生は、ビジュアルインジケータの始まりに対応する時間にセットされる。ノート（Notes）の場合には、ノート内のテキストを含むウィンドウが開けられる。そのような動作の発生は、図13で説明される。図13を参照すると、ビジュアルインジケータ1301がポイントされ、カーソル制御装置上のスイッチがクリックされた。これは、ミーティングの間に生成されたテキストを含んだテキストウィンドウ1302を開けさせる（ここでは、テキスト“ベティにより有用な指摘がなされた。上司とのミーティングにおいてこの議論を使う必要あり。”）。テキストウィンドウは、時間トラックディスプレイエリアの外の異なる位置で同様に現れる場合があることに注意すべきである。

【0113】タイムラインは、垂直或いは水平方向に表示することができ、または、連続したデータの長いストリングを示すために他の視覚化技術と組み合わせて使う

ことができることに注意すべきである。

【0114】プレーヤのフォームは、セッションの一時的データの、それが生成されたときと全く同じ形態で再生されることに限定されるものではない。そのようなプレーヤの例は、ミーティングプレーヤである。ミーティングプレーヤは、種々のミーティングダイナミックスを目につけるように表現するために使われる。これは例えば人が再生される声を認識することができないが、話を識別したいときに有用である。本発明のミーティングプレーヤは、ミーティングの間に発生する活動をシミュレートする。図14は、ミーティングプレーヤを表示するウィンドウ1400を説明する。図14を参照すると、ミーティングは、各々が、ミーティング参加者を表現するビジュアルインジケータ1401-1403を含み、ビジュアルインジケータ1404と1405はノートテキング装置（例えばラップトップコンピュータ）を表現し、ビジュアルインジケータ1406-1408は“ボタン”を表現する。再生の間、生じている現在のイベントを表現するアイコンはハイライトされる。ここでは、アイコン1401により表現された話者が話しているので、アイコン1401がハイライトされる。同様に、だれかがボタンを押したときに、対応するボタンアイコンはハイライトされて表示される。

【0115】話者のアイコンの中に話者のごく小さい画像を含めることは可能であることに注意すべきである。各々の関連づけられたアイコンが、タイムライン上の対応するトラックと同じ視覚的特徴を維持できることも同様に望ましい。そこで、もし、話者アイコン1401、ボタンアイコン1406、及び、コンピュータアイコン1404が、青の視覚的特徴を有するアダムと関連していれば、これらのアイコンは青となる。しかしながら、表示可能な色或いは他の要因の制限のために、これが不可能である状況もあり得ることが認められる。

【0116】更に、ミーティングプレーヤウィンドウに表示されているのが、LiveBoardページとタイトルインジケータ1410とクロック1411である。ページとタイトルインジケータ1410は、再生のこの瞬間にLiveBoardの上で現在見られている“ページ”を示す。クロック1411は、再生時間を示す。テーブル1409は、セッションがミーティングを表現するという事実に関して、ビジュアルキーを提供するために同様に表示される。

【0117】ミーティングプレーヤは、再生を異なる話者に変えるために使う事もできる。これは、ユーザが特定の話者によりミーティングを容易に横切ることを可能にする。異なる話者への切り替えは、所望の話者のアイコンに対して“ポイントしてクリックする”動作により達成される。これは、選択された話者に関して時間的に次の点へ再生をジャンプさせることになる。更に、この動作はあらゆるアイコンに対して作用し、そこで

は、選択されたアイコンと関連づけられたイベントの次のインスタンスへのジャンプが行われる（例えば、ページとタイトルインジケータ1410は、LiveBoardの次のページの始まりに再生をスキップするため、“ポイントしてクリックする”動作が行われる。）。このように、この例のミーティングプレーヤは、セッションの再生を制御するために更に動作する。【0118】ミーティングプレーヤは、分析することができる種々の他のイベントを示すように設計される場合がある。例えば、笑い或いは拍手がいつ生じているかを表すために、または、笑い或いは拍手の次の一時的な出来事へスキップするために、“笑い”アイコン1412或いは“拍手かっさい”アイコン1413を提供することができる。更に、好適な実施態様は、ミーティングを表現するために使われるが、同様の方法で、他の型の活動、例えば、プレゼンテーションを表現することができる。

【0119】現時点で好適な実施態様のLiveBoardウィンドウは、LiveBoardタイムストリーム、再生制御装置、及び、エディタのプレーヤとして同時に動作する。LiveBoardウィンドウは、前述のTivoliソフトウェアに基づく。LiveBoardウィンドウは、異なった再生モードで動作する。すなわち、過去の状態の正確な外観を再構築するアニメーションモード、カーソルがマーキング／編集が起きているエリアをポイントする“バウンシングーボール”モード、及び、“ナル（null）”モードである。アニメーションモードにおいては、LiveBoardウィンドウは、記録された順序で、LiveBoardタイムストリーム（すなわちタイムスタンプされた履歴リスト）を再生する。“バウンシングーボール”モードにおいては、セッションの終わり（またはページの終了状態）におけるLiveBoardの状態が表示され、オブジェクトに関係したイベントが起き、オブジェクトがハイライトされ、カーソルによりポイントされる。“バウンシングーボール”モードにおいては、カーソルの形はイベント（例えば、オブジェクトを生成するためのペン、オブジェクトを動かすための手、オブジェクトを削除するための消しゴム）の型を示すために変化する。

“ナル”モードにおいては、セッション（またはページの終了状態）の終りにおけるLiveBoardの終了状態が表示される。ナルモードは、LiveBoardウィンドウの更新が望ましくないときに使用される（例えば、それはユーザを迷惑させる）。

【0120】各々の再生モードにおいては、ゴーストと呼ばれる追加の特徴は、それが削除される後でも、ディスプレイ上にオブジェクトのかすかなバージョン（そのゴースト）を残す。この特徴は、ユーザによりオン・オフすることができる。ゴーストオブジェクトは、セッションの中へのインデックスとして使うことができる。

【0121】アニメーションモードとバウンシングーボールモードは、各々特定の時間に生じるイベントの型を示す。バウンシングーボールモードにおいては、カーソルはイベントの型を知らせるが、フルアニメーションモードにおいては、アニメーションが、それがどのようにアニメートされたかによりイベントの型を示す。

【0122】再生制御は、再生の間に表示されたオブジェクトとの相互作用により生じる。簡単な例として、ユーザはオブジェクトを選択して、LiveBoard上で再生コマンドを直接呼び出すことができる（例えば、LiveBoard上に書かれた時間）。他の再生制御の特徴は、以下に述べられる。

【0123】全てのLiveBoardの機能を利用することができますので、LiveBoardウィンドウは、エディタとして動作することができる。そこで、ユーザは、表示されたオブジェクトのそれら自身の操作を行なうことができ、或いは、図面を付加することができる。そのような操作は、それら自身のタイムストリームを発生させることになる。

【0124】LiveBoardウィンドウが、プレーヤと再生制御装置の両方として動作することは重要である。既知のノートテーキングシステムとは反対に、LiveBoardウィンドウは、ノートテーキング媒体の終了状態には厳密には基づかない再生制御手段を提供する。ユーザは、再生の間のあらゆる点でLiveBoardと相互に作用することができる。そこで、例えば、ユーザはセッションを再生して、セッションの先方の点にスキップし（例えばクロック上のポイント及びクリック動作により）、言葉がLiveBoardから消されたことに気付くことができる。これは、言葉の削除に関連している再生の一部を詳細に検討するためにユーザをトリガーすることができる。

【0125】図15は、現時点で好適な実施態様のLiveBoardウィンドウを説明する。図15を参照すると、LiveBoardウィンドウ1501は、セッションアクセスワークステーション上で実行されTivoliアプリケーションのためのユーザインターフェースのインスタンスである。LiveBoardウィンドウは、両側が、オブジェクトエリア1503に境を接する制御エリア1502から成る。制御エリア1502の種々の制御は、LiveBoardウィンドウ1501でアクティブである。アクティブで選択可能なオブジェクトは、オブジェクトエリア1503に表示されたマーキングである。同様に、再生ツール1504が説明される。再生ツール1504は、選択されたオブジェクトに関連がある点において、再生を始めるために使われる。

【0126】生成された履歴リストの故に、オブジェクトを選択して最も早い点を聞くためのLiveBoardとの相互作用の例が可能である。上述したように、履歴リストは実行されたプリミティブ動作のリストであ

り、各々のプリミティブ動作はタイムスタンプされ、オブジェクトと関連づけられる。オブジェクトは、典型的には、空間的に定義され、機能として実行された。例えば、点X1, Y1からX2, Y2へ線を引く。そこで、オブジェクトの選択がされるとき、空間的な位置を見て、その位置のオブジェクトを見つけ出し、オブジェクトに関連したプリミティブオブジェクトを見つけ出し、次いで、もっとも顕著なイベントの時間を選ぶことは簡単なことである。一般的な場合には、もっとも顕著なイベントは“生成”イベントである。しかしながら、他の時間がユーザーに興味を起こさせるかもしれない、ユーザーがもっとも顕著なイベント（以下に説明される）を選択できる選択動作が提供される。

【0127】LiveBoardタイムストリームは、概して二つの型、すなわち、単調と非単調に分類することができる。単調タイムストリームは、主に“生成”イベントを有する。起こる唯一の事は、新しいオブジェクトが生成される（移動、変更、消去はない）ことである。単調タイムストリームにおいては、圧倒的な大多数のオブジェクトは、それ（その生成イベント）と関連したたった一つのイベントを有する。

【0128】非単調タイムストリームは、生成、変更、移動、及び、削除イベントから成る。このように、各々のオブジェクトは、それに関連した一つ以上のイベントを有することができる。オブジェクトの位置が変化する変化（すなわち、移動）と、そうではない変化を区別することは有用である。もしタイムストリームが、どのような位置変化イベントも有していないければ、各々のオブジェクトはディスプレイの上で唯一の位置を有する。もし移動イベントがあれば、オブジェクトはセッションの間、多数の位置を占めるかもしれない。最後に、もしタイムストリームが削除イベントを有すれば、オブジェクトはセッションの間に、或る時間で消えるかもしれない。

【0129】Liveboardウィンドウは、スタンダローンの、或いは、タイムラインインターフェースと関連した再生制御装置として使用することができる。タイムラインインターフェースとの関連は、以下に述べられる。スタンダローンモードにおいては、単調タイムストリームについて、オブジェクトの選択及び再生ツール（play tool）の呼び出しは、オブジェクトについての“生成”イベントの時間における再生を引き起こすことになる（なぜなら、これはオブジェクトと関連している唯一のイベントであるからである）。

【0130】非単調タイムストリームの場合においては、問題は選択されたオブジェクトの再生時間を開始するために、何の“イベント”を使うかである。ユーザーが再生ツールでオブジェクトをポイントすると、システムは、オブジェクトと関連したもっとも早いイベントの時期のセッションを再生する。しかしながら、オブジェク

トについての他の時間が重要かもしれない。例えば、オブジェクトが移動された時間が重要かもしれない。オブジェクトについて他のイベント時間にアクセスする別の方法がある。例えば、オブジェクトと関連した全てのイベントを表にしたイベントメニューが提供される。ユーザーは、単に所望のイベントを選択する、再生はイベントと関連した時間で始まる。

【0131】別の技術は、再生時間を現在のオブジェクトに関連した次のイベントにスキップする次イベント再生ボタンを使うことである。タイムライン相関性を使う更に別の技術が以下に述べられる。

【0132】LiveBoard上の、また、Liveboardウィンドウ内のクロックの観念は、上述された。クロックは、Liveboardウィンドウで何時活動が起きたかの一時的な参照を提供するので、セッションの再生の際に有用である。クロックがクロックジェスチャーにより生成されたグラフィックオブジェクトであることを思い出すべきである。クロックは、全てのグラフィックアートオブジェクトと同じやり方で操作して編集することができる。しかし、クロックオブジェクトは特別であり、それは、それ自身の自分の“内部”時間を表現するそのデータ構造に、特別な特性を含む。再生ツールがクロックに触ると、再生点は、クロックの内部時間にセットされ、クロックと関連した幾つかのイベントの時間（例えばその生成時間）へセットされない。従って、クロックはそれ自身の履歴から独立した時間として取り扱うために使用することができる。デフォルトにより、クロックの内部時間は、クロックの生成動作と同じ時間である。しかし、そうである必要はなく、これは種々の有用な機能を導く。

【0133】クロックの内部時間は変えることができる。現時点で好適な実施態様においては、クロック上で編集ジェスチャー（現在、ポイントして保持している）が行われると、ユーザーがクロックの内部時間を変えることができるメニューがポップアップする。これらの変化は、相対的（例えば、内部時間をN秒だけ前方或いは後方へ移動する）、或いは、絶対的（すなわち、内部時間を特定の時間Tに移動する）のいずれかとすることができる。前者は、インデックスを付けるためにクロックの内部時間僅かに調整するときに一層正確であるので、もっとも有用である。

【0134】例えば、ミーティングにおいては、興味ある問題は取り上げられて議論される。LiveBoardのユーザーは、クロックを生成して、ノートを書くことにより、この討論にインデックスを付けることができる。しかし、通常、討論が始まってからある程度後に、クロックは生成される。従って、討論の開始に一層正確にインデックスを付けるために、その時間を僅かに逆方向に調整するために、クロックを編集することができる。この編集は、ミーティングの間、或いは、後でアク

セスセッションへの間にすることができる。後者は、セッティングが正確かどうかを判別するために、現在のクロックセッティングからミーティングを再生するのに便利であるので、もっとも有用である。現在の実施においては、編集が行われた後に再生を開始させて、ユーザに時間セッティングの妥当性をユーザに即時にフィードバックすることにより、ユーザに利便を計っている。

【0135】ユーザは、LiveBoardウィンドウ（以下、エディタとしてLiveBoardウィンドウを参照）経由でクロックを付加することによりアクセスセッションの間に、クロックを生成することができる。しかし、この状況においては、クロックの内部時間は、再生ツールの現在の再生点にセットされるものであり、アクセスセッションにおける（全く有用ではない）その生成時間にではない。例えば、一層早いミーティングをアクセスするユーザを考える。彼は、ミーティングのオーディオを聞き、ミーティングの間にインデックスを付けられなかった重要なアイディアを聞く。そこで、再生のこの点で、彼はクロックを生成し、次いでその隣にノートをタイプする。このクロックは、ミーティングへの新しいインデックスである。従って、オリジナルのミーティングにおける時間を保持するオブジェクトを生成することができるようになると、インデックス付加の質と量を、アクセス時間において強化することができる。

【0136】オブジェクトと関連した全てのイベントを見るることは、しばしば望ましい。例えば、ユーザは、オブジェクトの変更或いは削除の有無及び時期を知ることを望むかもしれない。上述したように、LiveBoardで選択したオブジェクトに関連した全てのイベントは、タイムラインインターフェースの選択トラックの上に表示される。これは、LiveBoardとタイムラインインターフェースの間の相関性により、ユーザが特定した関連があるセッションの中へのインデックスを表示するための手段を提供する。

【0137】LiveBoardウィンドウとの相互作用による相互関係の他の特徴は、図16-図20に関して述べられる以下のシナリオで説明される。LiveBoardウィンドウとタイムラインインターフェースの表現は、単純化されていることに注意すべきである。特に、LiveBoardウィンドウ上の制御エリアと、タイムラインインターフェースの概観時間トラックエリアは、図には示されていない。

【0138】図16を参照すると、ユーザはLiveBoardウィンドウ1601で言葉を見て、この言葉が書かれた時の会話の再生することを欲する。ユーザは、適切な選択技術（ここでは、ペン/マウスを使用して言葉を円で囲むジェスチャーにより）を利用して、言葉1602の“冗長”を選択する。次いで、タイムラインインターフェース1603は、選択された言葉と関連し

ている全てのイベントを示すために、選択トラック上で更新される。ここでは、ビジュアルインジケータ1604は、選択トラック1605に表示される。ビジュアルインジケータ1604は、言葉を書き込むイベントを表現する。この例では、他のイベントは、言葉“冗長”と関連づけられない。

【0139】同様に、タイムラインインターフェース1603で説明されているのは、再生時間インジケータ1606である。そこで、今度は、再生は言葉“冗長”が書かれた“後”的であることに注意すべきである。

【0140】図17は、LiveBoardウィンドウとタイムラインインターフェースの間の同期を説明する。ユーザは、タイムラインインターフェース1702内の再生時間インジケータ1703をビジュアルインジケータ1604の直前に移動する。次いで、LiveBoardウィンドウ1701は、ちょうどその時にLiveBoardの上にあったものを表示するために、更新される。特に、図16と比較された時、言葉“冗長”はLiveBoardウィンドウに現れないことに注意すべきである。

【0141】再生コマンドが呼び出されると、LiveBoardウィンドウとタイムラインインターフェースの両方は、同期して更新される。しばらくして、言葉“冗長”は、LiveBoardに現れる。

【0142】ここで、ユーザは、選択トラックの上のイベントの直前に、タイムラインの上にいくつかのイベントがあったことに気がつく。ユーザは、タイムラインの上のこれらのイベントを表現するビジュアルインジケータを選択し、すぐに、図面の対応している要素はLiveBoardウィンドウでハイライトされる。すなわち、図面とタイムラインの相互関係は双方向性である。

【0143】これは、図18で説明される。ビジュアルインジケータ1804は、タイムラインインターフェース1803から選択される。LiveBoardウィンドウ1801において、言葉“ルーター(route)" 1802はハイライトされる（ここでは点線の円で示した）。従って、ビジュアルインジケータ1804に対応するイベントは、言葉“ルーター”と関連づけられる。

【0144】次に、ユーザは、LiveBoardウィンドウの図の上で太いボックスを見る。そして、ボックスを選択する。三つのビジュアルインジケータは、タイムラインインターフェース（または、三つのビジュアルインジケータは、編集トラック上でハイライトすることができる）の選択トラック上に表示される。第1の型のビジュアルインジケータは、ボックスのための生成イベントを示し、第2の型のビジュアルインジケータは、ボックスが太くされた時にライン幅変更イベントを示し、第3の型のビジュアルインジケータは、図面上のボックスの移動のイベントを示す。上記されたように、タイム

ラインインターフェースの上のビジュアルインジケータは、それらが表現するイベントの種類を示すために視覚的に区別される。ユーザは、ボックスが太い理由に興味があり、時間マーカーをライン幅変更イベントへ移動し、再生ツールを使って再生を呼び出す。

【0145】これは、図19で説明される。ボックス1902は、LiveBoardウィンドウ1901でハイライトされる。ビジュアルインジケータ1904-1906は、タイムラインインターフェース1903でハイライトされる。上述したように、これらのビジュアルインジケータの各々は、オブジェクトと関連したイベントの特定の型を表現する。更に、現時点で好適な実施態様においては、各々のビジュアルインジケータは、イベントの型へ対応する或る視覚的な特性、典型的には色を有する。

【0146】ユーザは、どちらから再生を開始するかビジュアルインジケータ1904を選択する。

【0147】次に、ユーザは、図面の別のページの上の図を考える。この図は、ミーティングの間に、何度も描かれ、消され、書き直されたものであり、ユーザは、以前のバージョンが正しいと思われなかった理由を聞きたい。ユーザは、図が位置している図面のエリアを選択する。ここで、タイムラインの幾つかのセクションがハイライトされ、削除或いは移動されたオブジェクトを含む、図のそのエリアにあった全てのオブジェクトについてのイベントが明らかになる。ユーザは、時間マーカーを、第1のハイライトされた削除のビジュアルインジケータに移動し、なぜ第1のバージョンが変更されたかの議論を見つける。

【0148】特定の位置でオブジェクトがなる、或いは、なったビジュアルキーは、ゴーストオブジェクトである。ゴーストオブジェクトは、セッションの間の或る時間の点において、オブジェクトがこの特定の位置に存在したこと示すために使用されるオブジェクトのかすかな輪郭である。ユーザは、種々のオブジェクトゴーストに気がつくことができ、そこに何が有ったのか、或いは、有ることになるのかを知ることを欲する（例えば、セッションの終わりにおいて項目が消されたことがゴーストオブジェクトにより明らかになる項目のリスト）。

【0149】これは、図20で説明される。図20を参照すると、ユーザは、LiveBoardウィンドウ2001の図のエリア2002を選択した。簡単に図16を参照すると、エリア2001は、言葉のリスト、“付加”、“ルーター”、及び、“冗長”を含むエリアに対応する。タイムラインインターフェース2003は、ここでハイライトされた多数のビジュアルインジケータ2004-2006を有する。これらは、エリア2002と関連したイベントに対応するイベント、すなわち、言葉“付加”、“ルーター”、及び、“冗長”に関連する

イベントに関係する。これは、単に空間的な相関であること、すなわち、リストから消された言葉を表現するイベントも同様にハイライトされたことに注意すべきである。更に、図20に説明されているのは、“ゴースト”オブジェクト2007である。再生のこの点で、“ゴースト”オブジェクト2007により示された位置のオブジェクトは2007は消された。

【0150】上述したように、エディタは、ユーザが事実の後にタイムストリームとイベントデータを付加することを許す。LiveBoardウィンドウがエディタとしていつ使われるか、何が編集されるかは、LiveBoard（または、付加された新しいマーキング）の上のマーキングである。この場合には、LiveBoardタイムストリームへの付加及び／又は削除は、そのような編集の結果として行われる。また、別のタイムストリームを生成することもできる。

【0151】エディタとして使われるとき、LiveBoardウィンドウは、非参加者が記録されたセッションを効率的に検討することができる非常に有効な手段を提供する。これは、部分的には、LiveBoardの全ての基本的な能力がセッションの再生の間利用することができるからである。これは、編集セッションの前後の両方のセッションの終わりにおけるLiveBoardウィンドウを示す図21と図22に関して、もっともよく述べられる。この例のLiveBoardは、バウンシングボール再生モードで動作している。

【0152】図21を参照すると、種々の手書きのマーキングがLiveBoardウィンドウ2101に現れている。上述したように、各々のこれらの手書きのマーキングは、セッションの中へインデックスを付けるために使うことができる。しかしながら、或るミーティングの間、マーキングのいくつかは無関係（例えば、マーク2102）であったり、判読し難い（例えば、マーク2103）場合がある。更に、議論された重要な点があるがLiveBoardに対してはなにも成されていないような場合もある。

【0153】図22は、だれかが編集した後の、図21のLiveBoardウィンドウを説明する。図22を参照すると、編集されたLiveBoardウィンドウ2201は、エディタにより成された注記に適応させるために拡張された。編集されたLiveBoardウィンドウ2201はクロックを使用してテキストを挿入し、一層多くのインデックスと情報を提供する。図22において、何がその点で議論されているかを表すために、テキストは各々のクロックと関連づけられた。そこで、例えば、テキスト2203は、マーク2103が何を意味するかを表すために使うことができる。クロック／テキスト2202は、マーキングの収集が何を表すか、そして、何時マーキングの生成が始まったかを表す。明らかに、編集されたLiveBoardウィンド

ウは、次の視聴者が再生制御装置としてLiveBoardウィンドウを一層効果的に使うことを可能にする。

【0154】エディタとして、LiveBoardウィンドウの上に存在するマーキングを変えることができる。これは、例えば、ウィンドウの次の視聴者の注意がそらされないように、無関係なマーキング、例えば、図21のマーキング2102を消すために行うことができる。

【0155】タイムストリームアーキテクチャの別の局面は、獲得されたタイムストリームに対する制御である。ユーザは、獲得されたタイムストリーム、特に、オーディオとビデオのように“文字通り”的記録であるタイムストリームに非常に関心がある。彼らは、彼らの活動及び会話の文字通りの記録に誰がアクセスしたかに関心がある。彼らは、彼らが何時記録されたを知ることに関心があり、彼らはこれに対する制御を望む。他方では、興味はあるが記録されなかった議論の後で、ユーザはし彼らがそれを記録していたらとばしば望む。

【0156】ユーザ制御は、タイムストリームアーキテクチャでサポートされる。しかしながら、実施は、セッションのキャプチャの間とその後の両方で、ユーザに制御を効果的に与えるためのユーザインターフェースツールを必要とする。

【0157】タイムストリームアーキテクチャは、“制御イベント”的特別な取扱いの提供する。制御イベントは、以下のものを指定する

一時間スパン（時間と期間を開始させる）

一制御すべきタイムストリーム

一制御の種類（保護のページレベル）

一制御が有効になるとき

【0158】獲得されたセッションが多数のタイムストリームを有することができるとすると、制御イベントはそれらを選択的に制御することができる。例えば、議論の微妙なスパンの間のオーディオだけを制御することが有用である。制御イベントは、セッションの全てのタイムストリームを特定することもできる。

【0159】いくつかの種類の制御がある。記録された資料を、データベースから物理的にバージすることができる。記録された資料を聞くか見ることへのアクセスは、拒否したり、特定のユーザ或いはユーザのグループに与えることができる。アクセスは、読むことのみが可能なアクセス、新しい資料をアペンドすることが可能なアクセス、資料を完全に編集することが可能なアクセス、或いは、アクセス制御を変更することが可能なアクセスのような、異なる型とすることができます。例えば、制御イベントは、ミーティングの議長だけがミーティング記録を編集することができ、グループのマネージャーだけがアクセス制御を変えることができ、ミーティングの参加者だけが新しい資料を獲得された資料にアペンドすることができ、特定の組の他の個人のみが獲得さ

れたオーディオを聞くことができるように特定することができます。

【0160】最後に、制御のタイミングを指定することができる。これは、ページするためにもっとも有用である。獲得された記録は、明細を記すと直ちに（例えば、総員の議論を消すために）、セッションの直後に（例えば、興味をそらさないミーティングを消すために）、或いは、何日か何ヵ月後に（例えば、文書整理と保管管理のために）のように、異なった時間にページすることができる。しかし、非参加者に対してはミーティング記録へのアクセスを1週間拒否して、議長に記録を編集するという時間を与えるように指定することもできる。

【0161】タイムストリームアーキテクチャは、制御イベントが種々の制御を指定するのを可能にするが、アーキテクチャはそれらを直接には実施せず、アーキテクチャの種々の構成要素への制御を代表する。指定の多くは、それらを実施するために、データベースでコード化することができる。タイムストリームプレーヤとタイムラインインターフェースも、同様に或るアクセス制御を実施する補助となる。ページすることは、レコードにより、または、特定のタイムストリームのデータフォーマットを理解しているなんらかのプロセスにより通常実施されなければならない。ページするプロセスは、ページすべきデータを重ね書きするか（“ゼロで埋める”）、或いは、実際に再度書き込んでページすべきデータを外部に出したままにしておくことにより記録を圧縮するかのいずれかとすることができます。タイムストリームアーキテクチャは、それ自身は制御イベントの完全な実施を保証しないが、与えられた状況の中で利用可能な構成要素により行うことができる最善のことを行う。

【0162】時間に基づいた方法でアクセス制御を提供することは、一般に非常に微妙であることが判っている。例えば、微妙な決定を取り扱っているので、或る10分のスパンの議論のオーディオ/ビデオ記録をページすることを考え、また、ビデオ記録が壁のホワイトボードを示していると考える。たとえ、10分のスパンがビデオからページされるとしても、そのスパンの後に、その討論の残部がホワイトボードの上に残らないという保証はない。しかしながら、実際的な問題として、時間に基づいた制御は、殆どの時間で有用である。

【0163】制御イベントは、ユーザがそれらを容易に生成して編集することができる場合にのみ有用である。ユーザにとっては、セッションの後にタイムラインインターフェースで制御イベントを指定することが自然である。

【0164】現時点で好適な実施態様においては、ユーザにセッションの間のオーディオの記録に対する控え目の制御を与えるために、ユーザインターフェースツールの簡単なセットが提供される。ユーザに何時記録が開始されたかを気付かせることは重要である。ビデオカメラ

の上の赤い光がこれを行う。Liveboardについては、記録が行われている間に、点滅する“記録中”が表示される。両方とも物理的な装置の上のハードボタンであるボーズボタン及び記録ボタンと、LiveBoard或いはコンピュータディスプレイの上のソフトボタンにより、ユーザがレコーダをオン・オフすることができる。しかし、この制御は予期されなければならない。すなわち、ユーザは、記録またはボーズのいずれとするために、興味があるか微妙であるかのいずれかであるかを前もって知っていなければならない。実際には、これは有効ではない。すなわち、それらが始まってから十分あとになるまでは、ユーザはこれらの状況をしばしば認識できない。従って、ポストホック(post-hoc)制御が必要とされる。

【0165】我々のユーザインターフェースは、“記録オン”と“記録オフ”(ボーズと記録に加えて)の概念を実施することにより、これを提供するものであり、これは、セッション中に適当な方法により、ユーザが事実の後の記録を制御することを可能にするものである。このアイディアは、全体のセッションが実際に記録されることであり、ユーザは、セッションの部分に記録オンである或いは記録オフであるかをマークすることができ、セッションの後で記録の記録オフの部分はページされる。

【0166】セッションが記録するために開始させられると、点滅する“記録オン”注意が、記録オフボタンと一緒に表示される。記録オフボタンが押されると、ユーザは、現時点で或いはN分前の時点で、セッションに記録がオフであることをマークすることができる。例えば、もし議論が微妙な話題に移って、討議者が数分後にノートに書き込むと、彼らは、記録オフとなる最後の、例えば、5分間にセッティングすることができる。同様に、セッションが記録されていないときには、点滅する“記録オフ”が、記録オンボタンと一緒に表示され、これにより、ユーザがセッションに記録がオンであることをマークすることができる。例えば、討議が面白くなつたときに、記録の上に最後の10分を元へ戻すことができる。これらの制御は、ユーザが全体のセッションを記録すべきか否かを決定することができるセッションの終わりへ固執する。

【0167】これらのユーザインターフェースツールが行うこととは、セッションの間に制御イベントを生成させることである(記録オフセグメントはページされたとしてマークされ、記録オンセグメントはアクセスを許可するとしてマークされる。)。セッションが閉会した後、プロセスは、矛盾する指定(すなわち、セッションのスパンが記録オンとオフの両方にマークされている)について作りだされた制御イベントを分析するプロセスが実行され、次いで、記録をページするためにレコーダがコールされる。対立解決のための一つの技術は、一層最近

の制御イベントに優先権を与えることである。しかしながら、ユーザは、タイムラインインターフェース経由で制御イベントを見て、ページをコールする前にそれらを編集するように頼むことができる。

【0168】このユーザインターフェースの一つの困難な点は、記録を変えるために何分戻るのかを知ることがユーザにとって困難であることである。これに対する一つの改良は、ユーザに一層意味がある時間点を与える我々のオーディオ分析ツールを使うことだろう。例えば、記録オフボタンを押した後に、最近のオーディオのボーズに対するバックアップ点或いは話者の変化のメニューを現すことができる。ユーザは、正しい時間点を決定するために、これらから再生することができた。

#### 【図面の簡単な説明】

【図1】 本発明の現時点での好適な実施態様における一時的データの捕獲及び再生のためのシステムのブロック図である。

【図2】 本発明の現時点での好適な実施態様において利用することができるセッションアクセスワークステーションのブロック図である。

【図3】 本発明の現時点での好適な実施態様のセッションアクセスワークステーションの動作の単純化された疑似状態である。

【図4】 本発明の現時点での好適な実施態様のセッションアクセスワークステーションに連結されるディスプレイのビジュアルユーザインターフェースの図である。

【図5】 本発明の現時点での好適な実施態様のセッションのためのタイムラインユーザインターフェースディスプレイの図である。

【図6】 概観時間トラックディスプレイエリアと焦点時間トラックディスプレイエリアの関係をハイライトするタイムラインインターフェースの単純化された図である。

【図7】 図6の焦点バーが引き伸ばされこれに従って焦点時間トラックディスプレイエリアの寸法が変えられている動作の結果を示す図である。

【図8】 二つのセッションの再生を制御するためのタイムラインユーザインターフェースディスプレイシステムの図である。

【図9】 二つの不連続な焦点セグメントが概観時間トラックエリアに表示され、焦点時間トラックエリアが各々の焦点セグメントの詳細を含むタイムラインインターフェースの図である。

【図10】 図9の焦点時間トラックエリアのサイズ修正の結果を示すタイムラインインターフェースの図である。

【図11】 LiveBoard ウィンドウの選択動作の結果として生じたトラックセクションを含む複数のキャプチャ装置で記録したセッションの再生のためのタイムラインインターフェースの例である。

【図12】 一つのトラックの上で図11のオーディオのトラックを併合した結果を示すタイムラインユザインターフェースディスプレイの更なる図である。

【図13】 ノートトラックの上にノートを開けた結果を示す図11のタイムラインユザインターフェースディスプレイの更なる図である。

【図14】 図11のタイムラインユザインターフェースを生成するのに使用した同じセッションについてのミーティングプレーヤの図である。

【図15】 本発明の現時点での好適な実施態様におけるLiveBoardウィンドウの図である。

【図16】 LiveBoardウィンドウとタイムラインインターフェースの間の相関性の図である。すなわち、グラフィックオブジェクトからタイムラインへのインデックス付けである。

【図17】 Liveboardウィンドウとタイムラインインターフェースの間の相関性の図である。すなわち、LiveBoardウィンドウとタイムラインインターフェースの両方が同時に同じものを反映する同期化されたディスプレイ状態である。

【図18】 Liveboardウィンドウとタイムラインインターフェースの間の相関性の図である。すなわち、一時的なものから空間へのインデックス付けである。

【図19】 LiveBoardウィンドウとタイムラインインターフェースの間の相関性の図である。すなわち、グラフィックオブジェクトにつき多数のイベントである。

【図20】 LiveBoardウィンドウとタイムラインインターフェースの間の相関性の図表である。すなわち、オブジェクトエリアの位置選定によってインデックスを付ける。

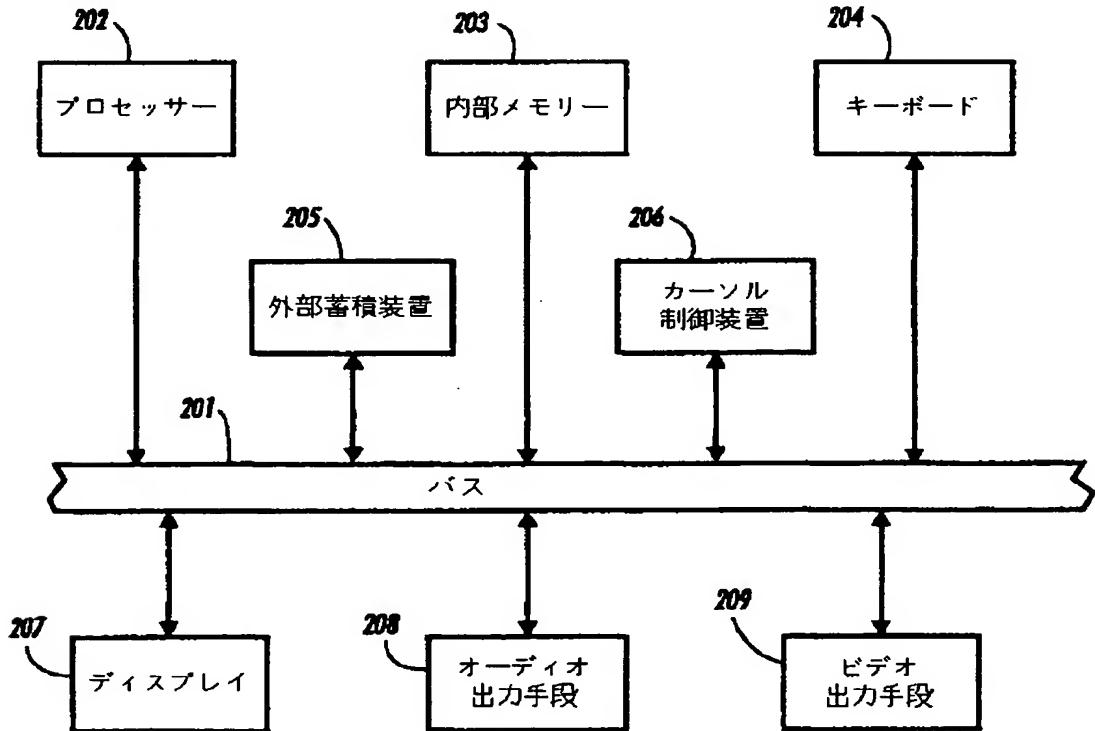
【図21】 LiveBoardウィンドウが編集された前の“バウンシングボール”再生モードにおけるLiveBoardウィンドウの例を提供する。

【図22】 LiveBoardウィンドウが編集された後の“バウンシングボール”再生モードにおけるLiveBoardウィンドウの例を提供する。

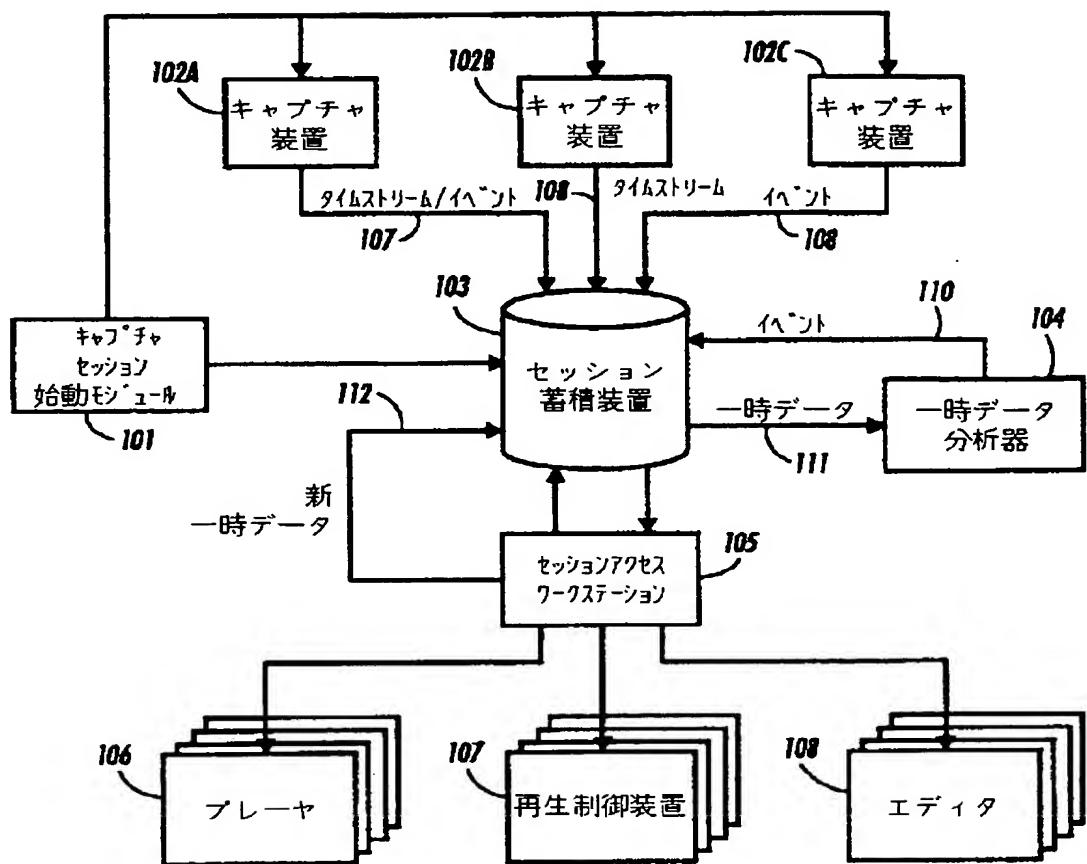
【符号の説明】

101 キャプチャセッション始動モジュール、102a キャプチャ装置、102b キャプチャ装置、102c キャプチャ装置、103 セッション蓄積装置、104 一時データ分析器、105 セッションアクセスクーステーション、106 プレーヤ、107 再生制御装置、108 エディタ、107タイムストリーム/イベント、108 タイムストリーム/イベント、109 タイムストリーム/イベント、110 イベント、111 一時データ

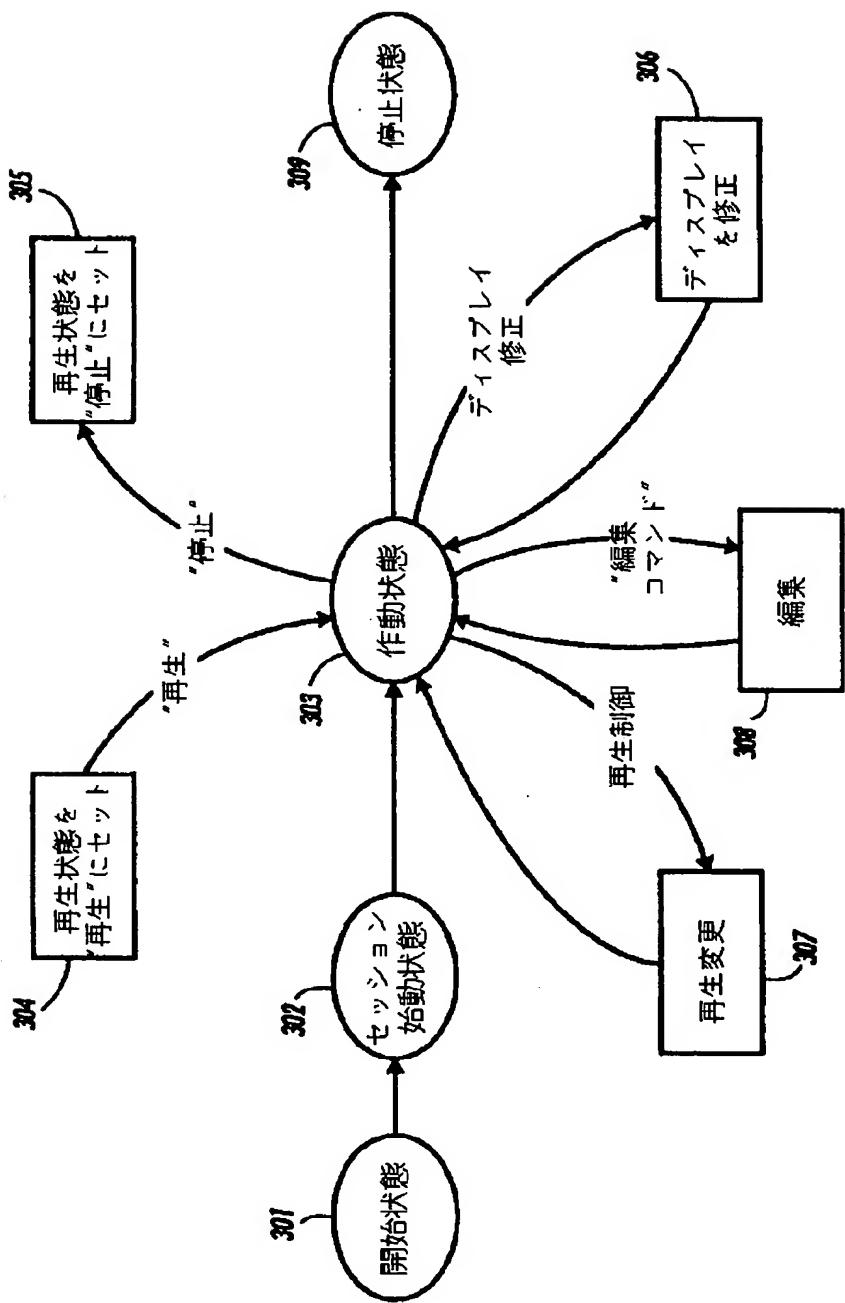
【図2】



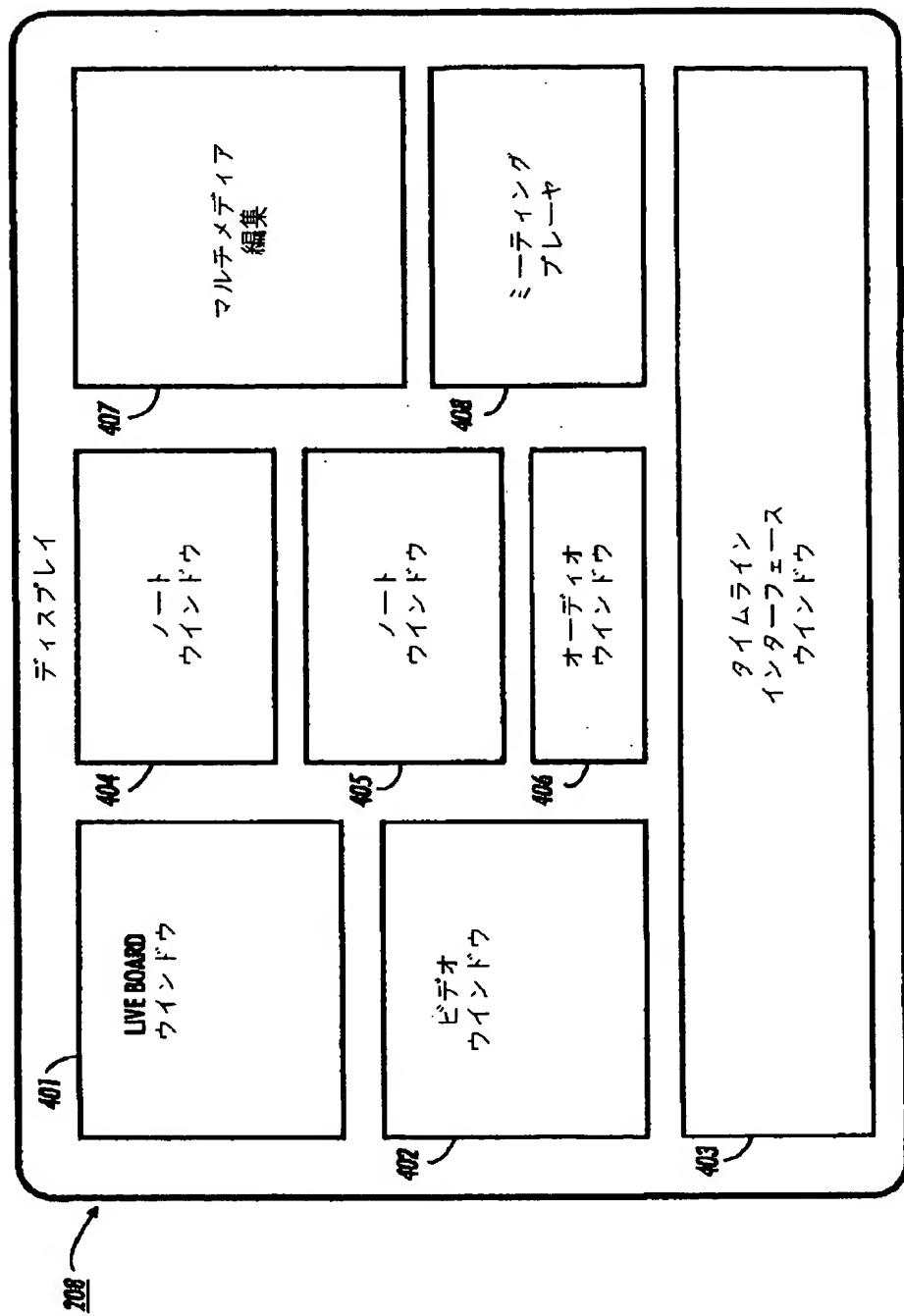
【図1】



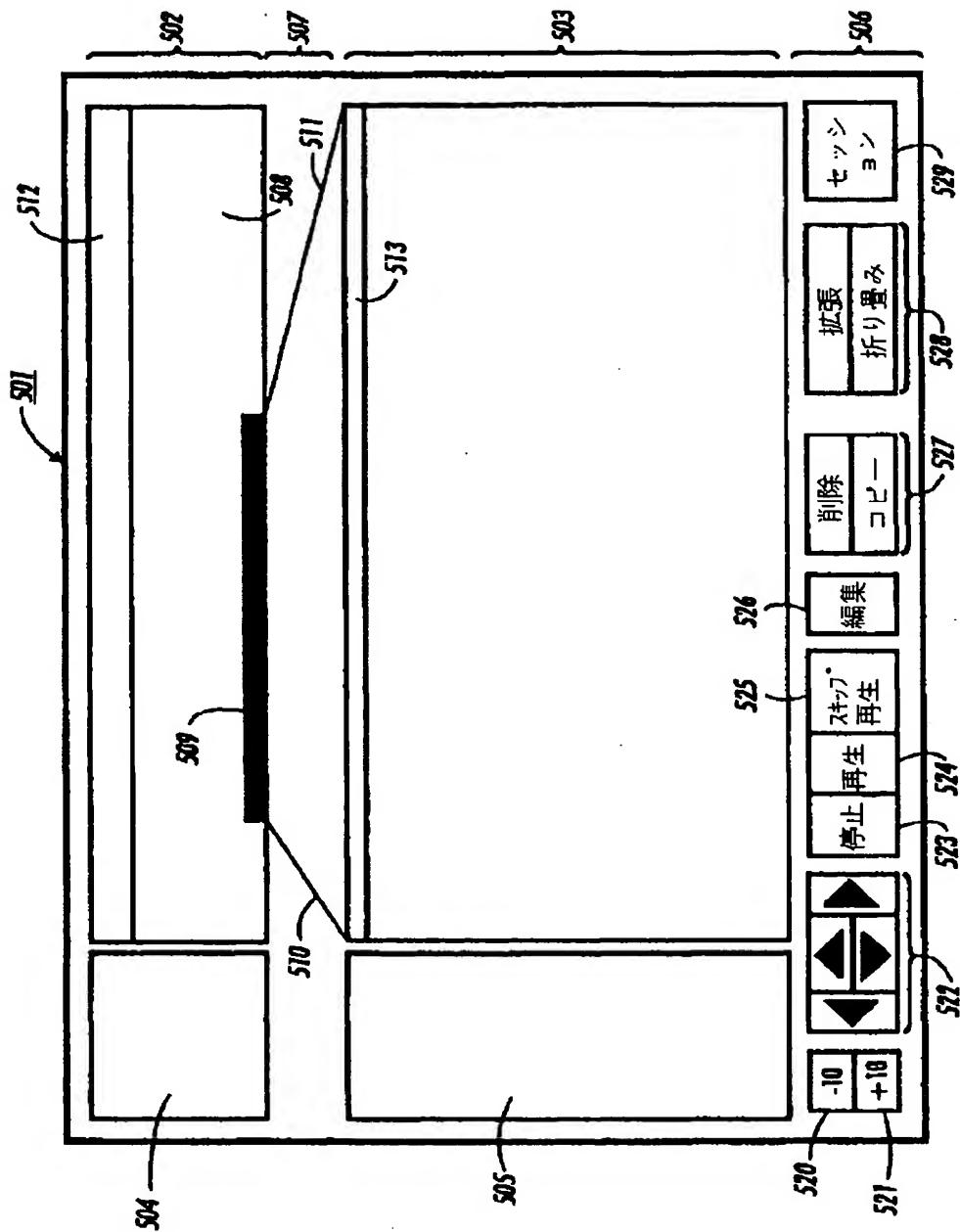
【図3】



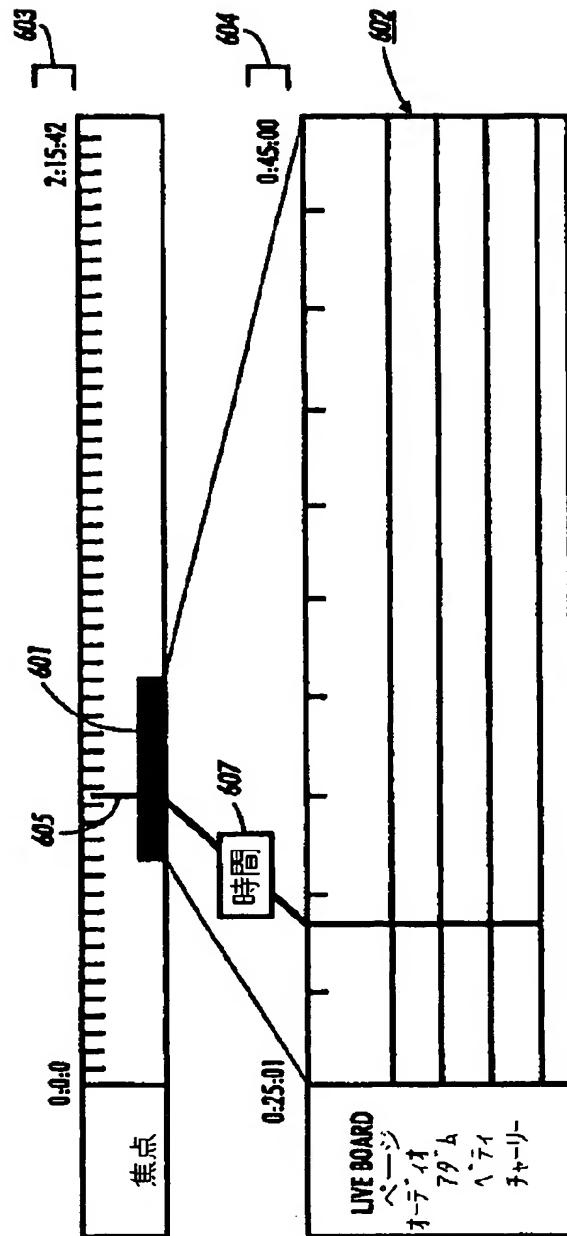
【図4】



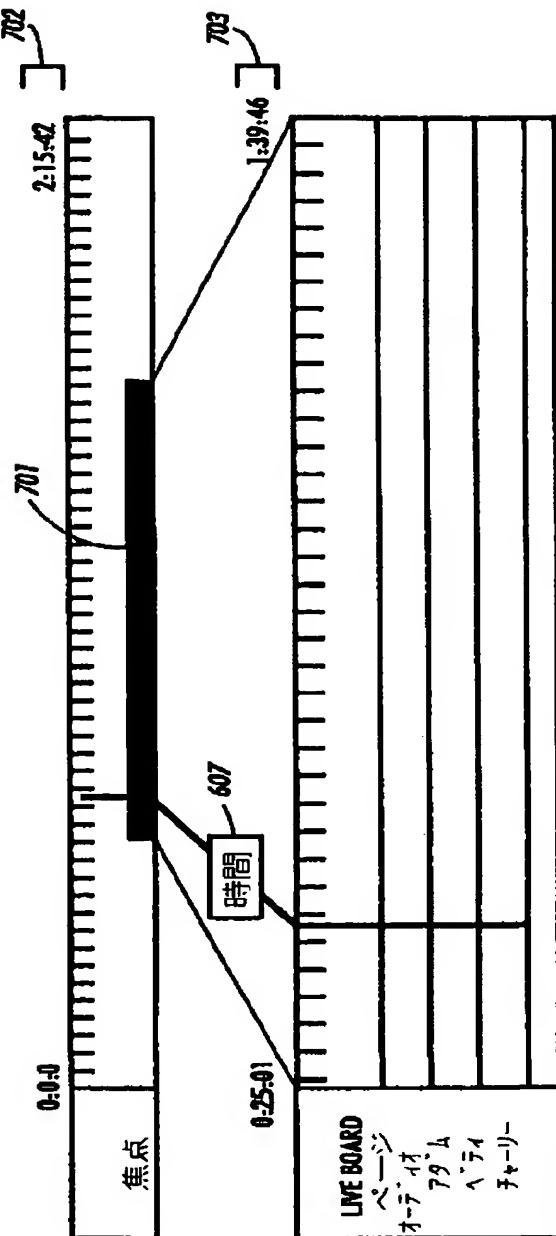
【図5】



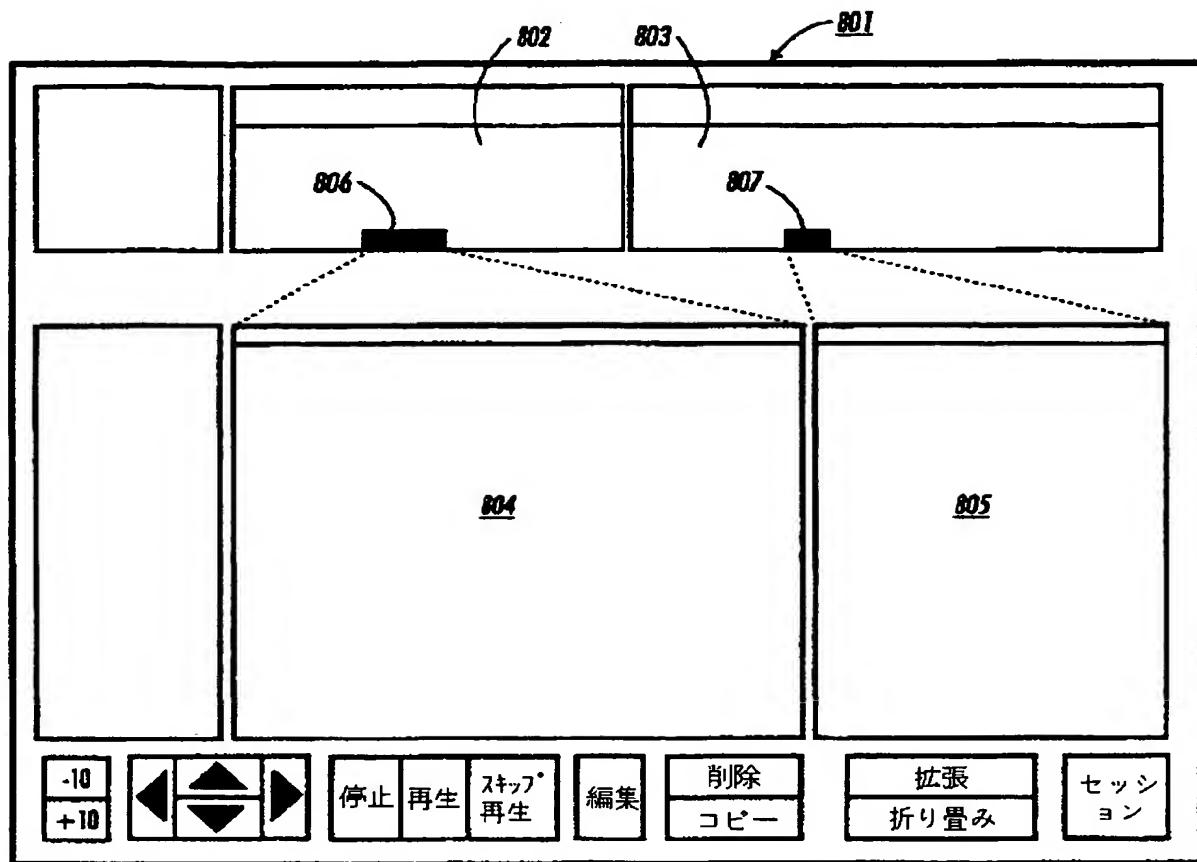
【図6】



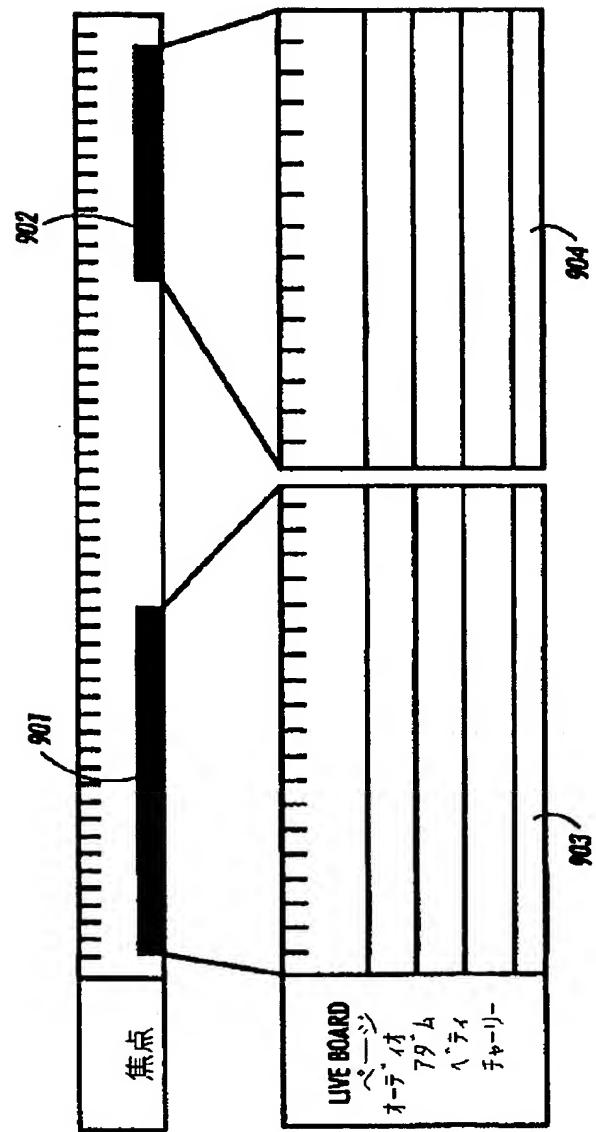
【図7】



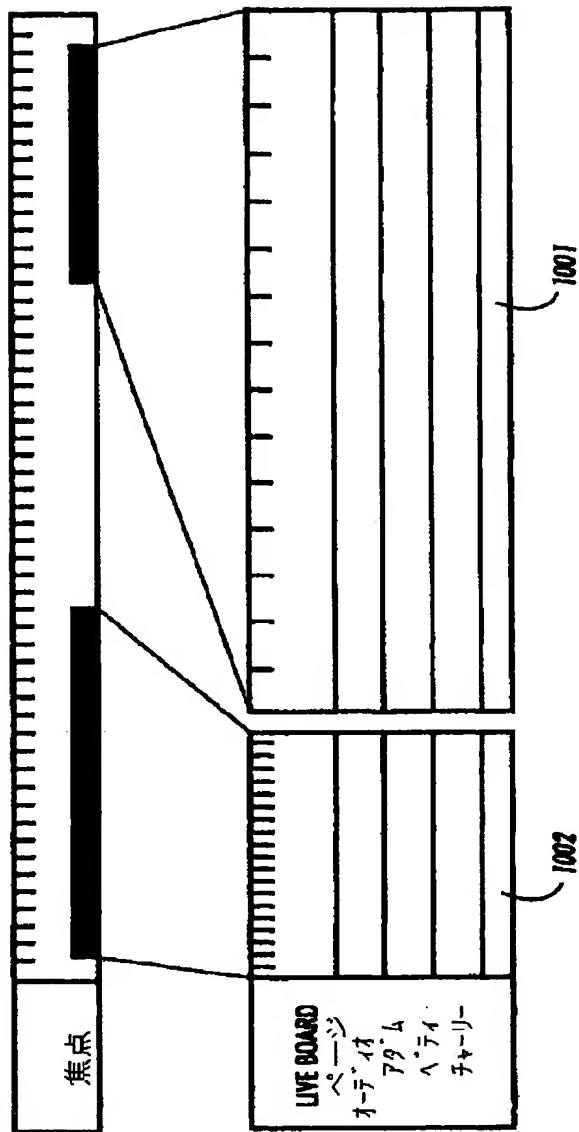
【図8】



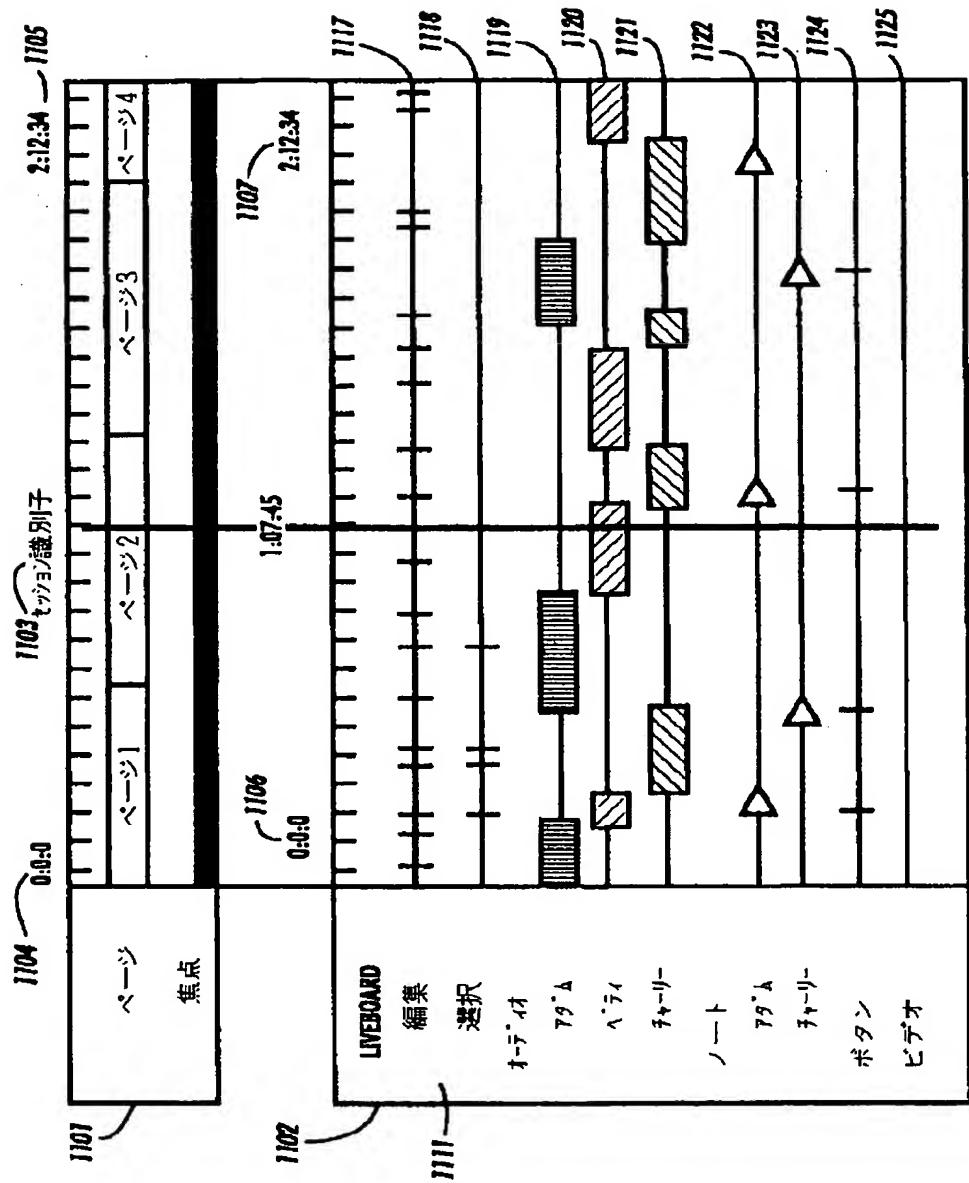
【図9】



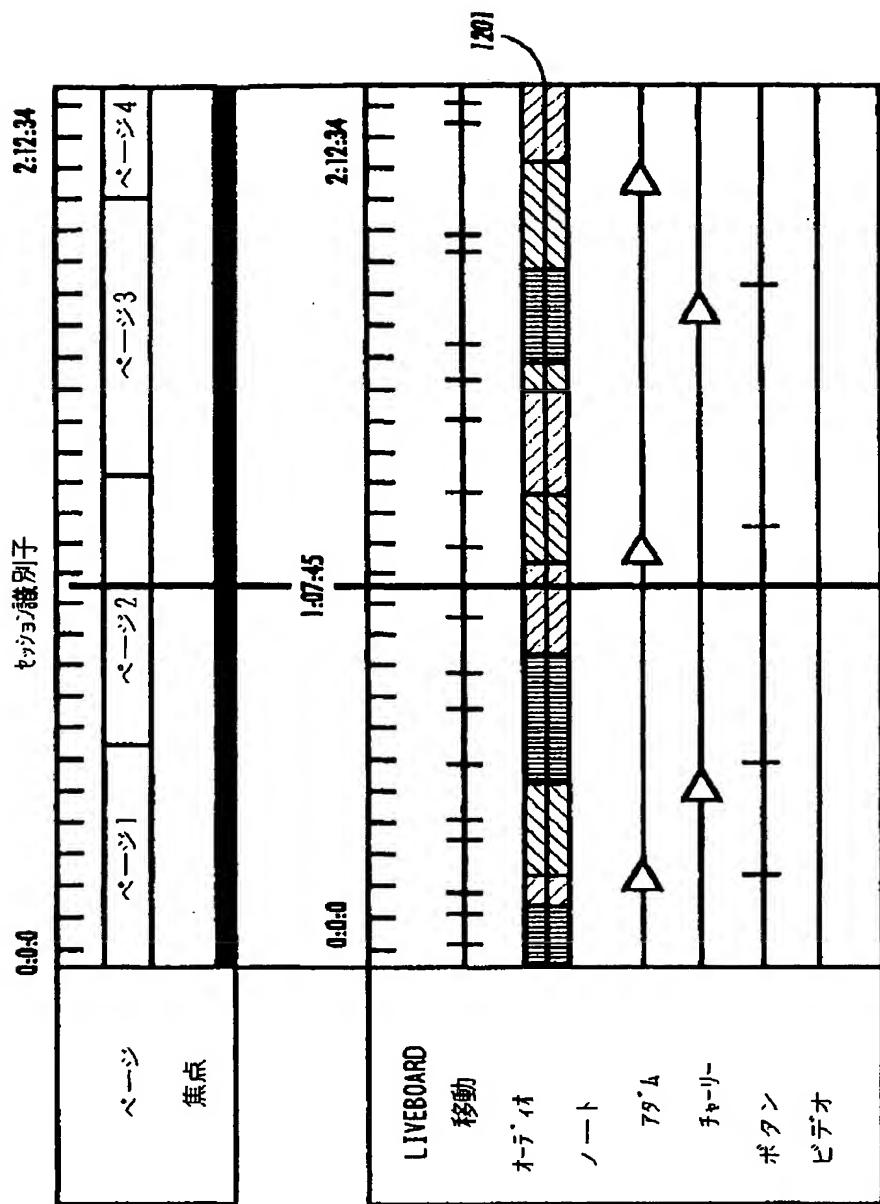
【図10】



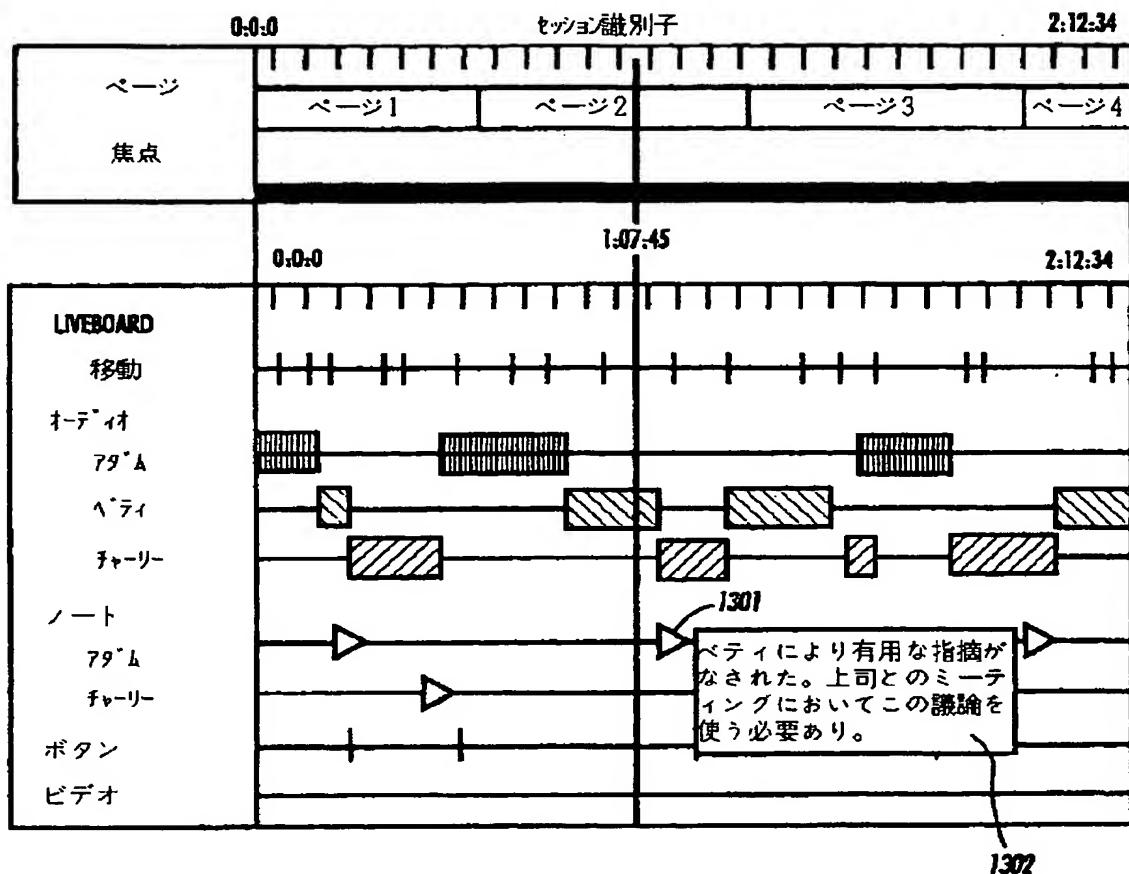
【図11】



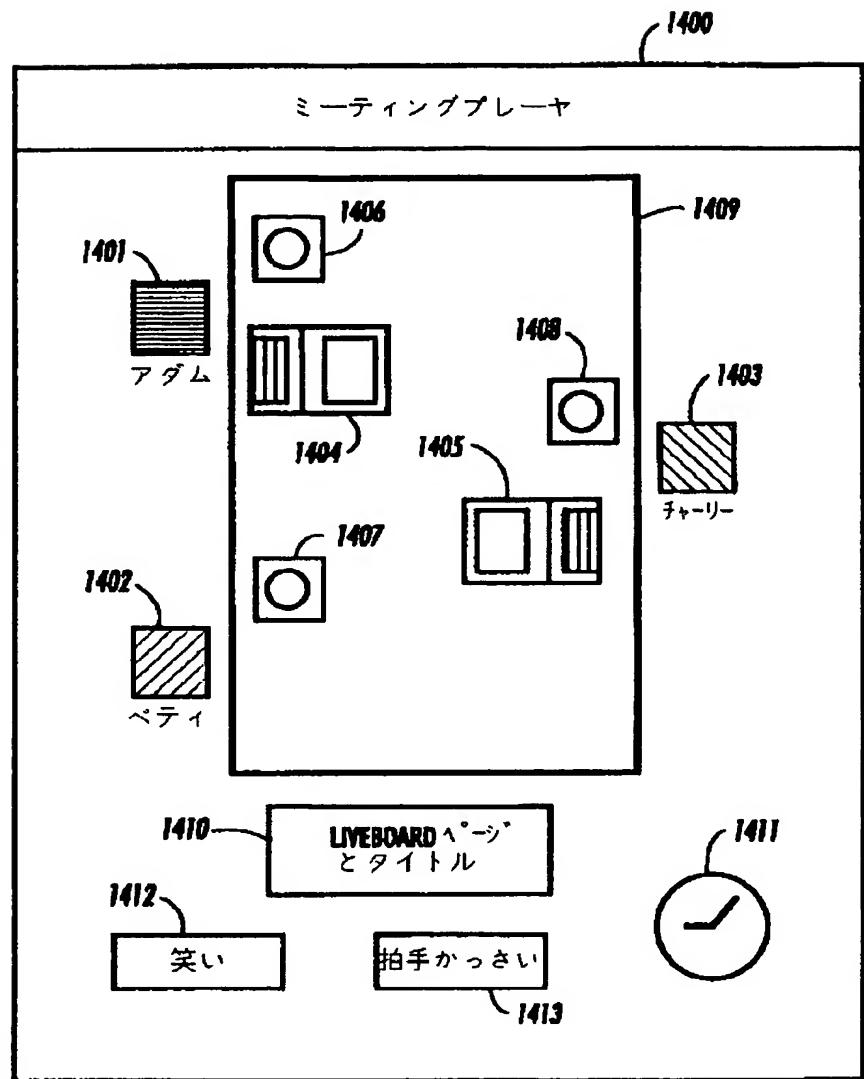
【図12】



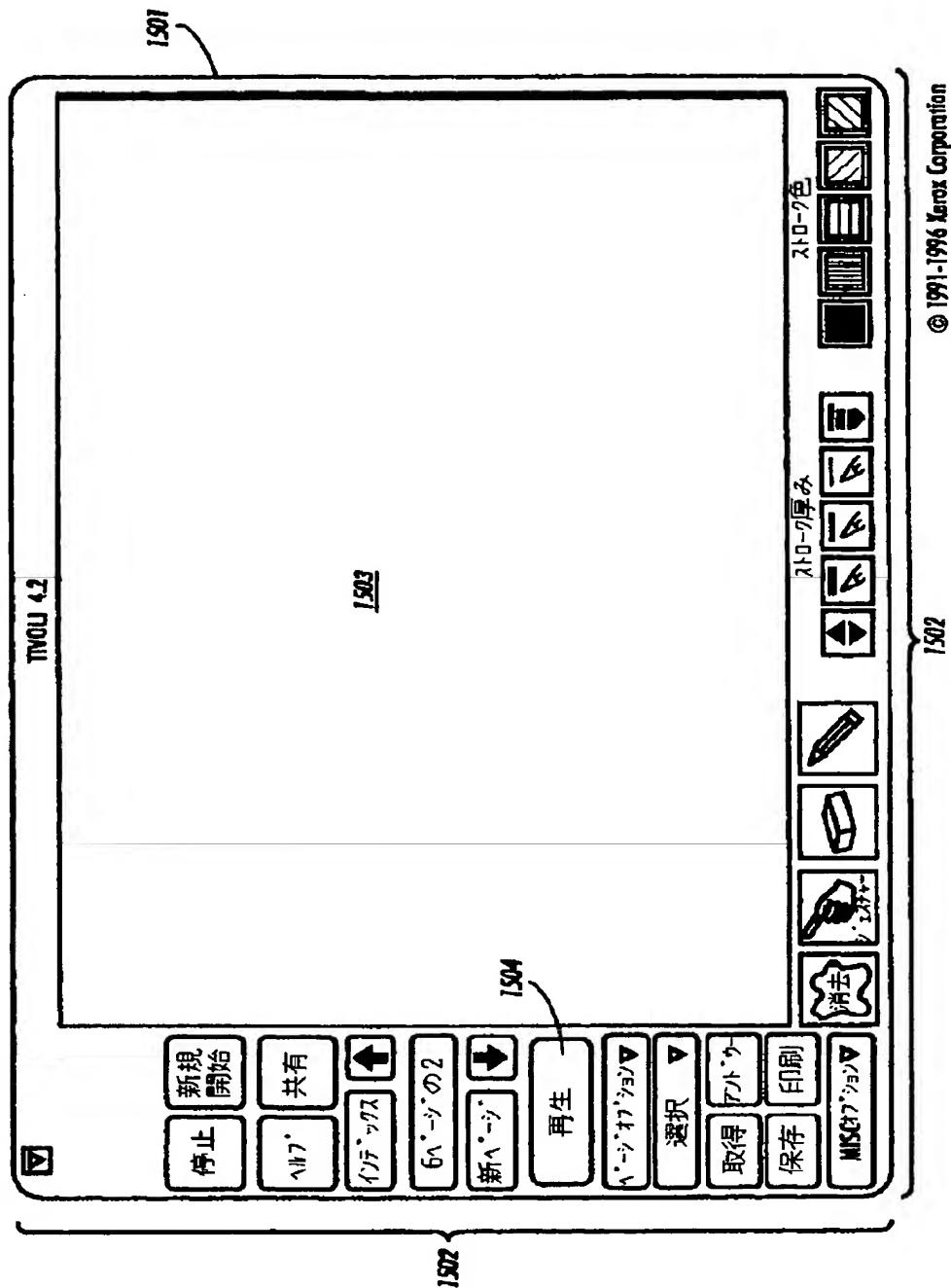
【図13】



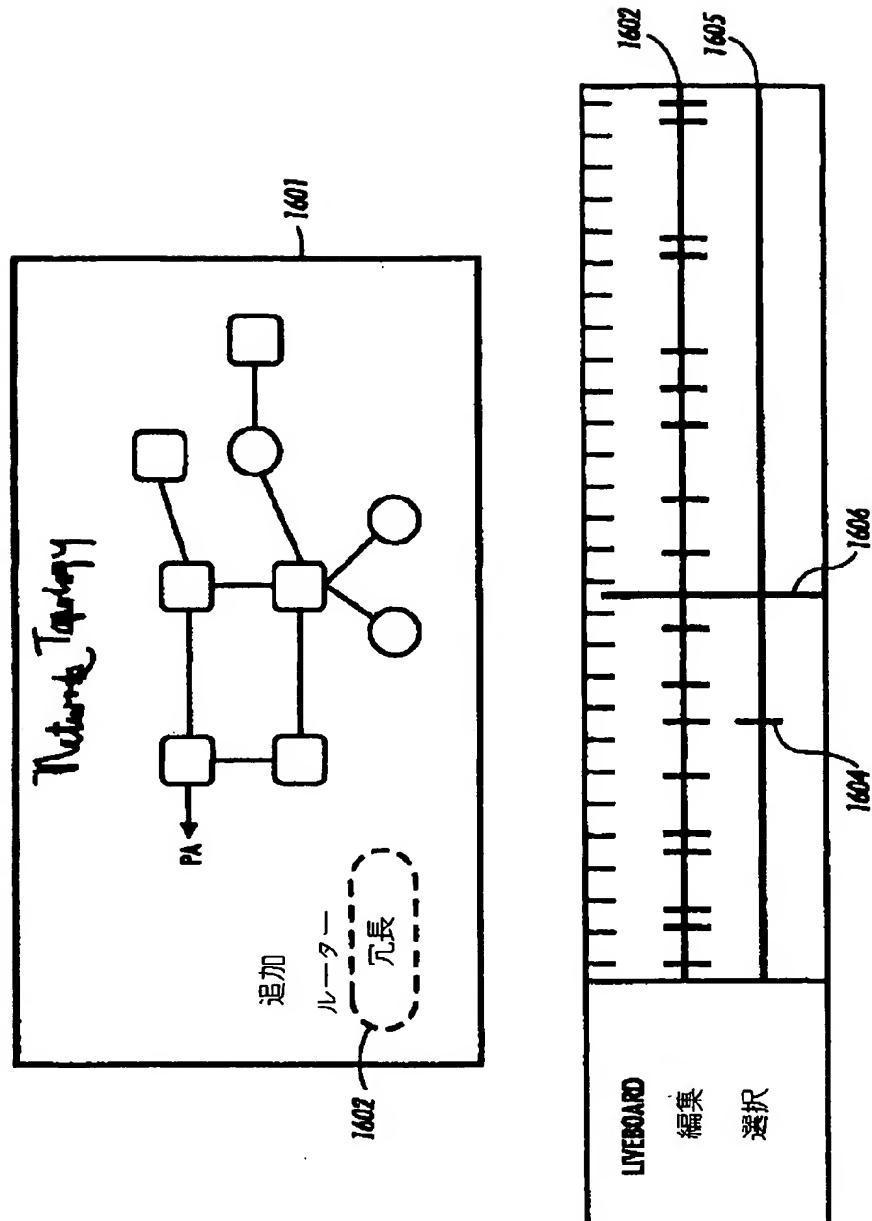
【図14】



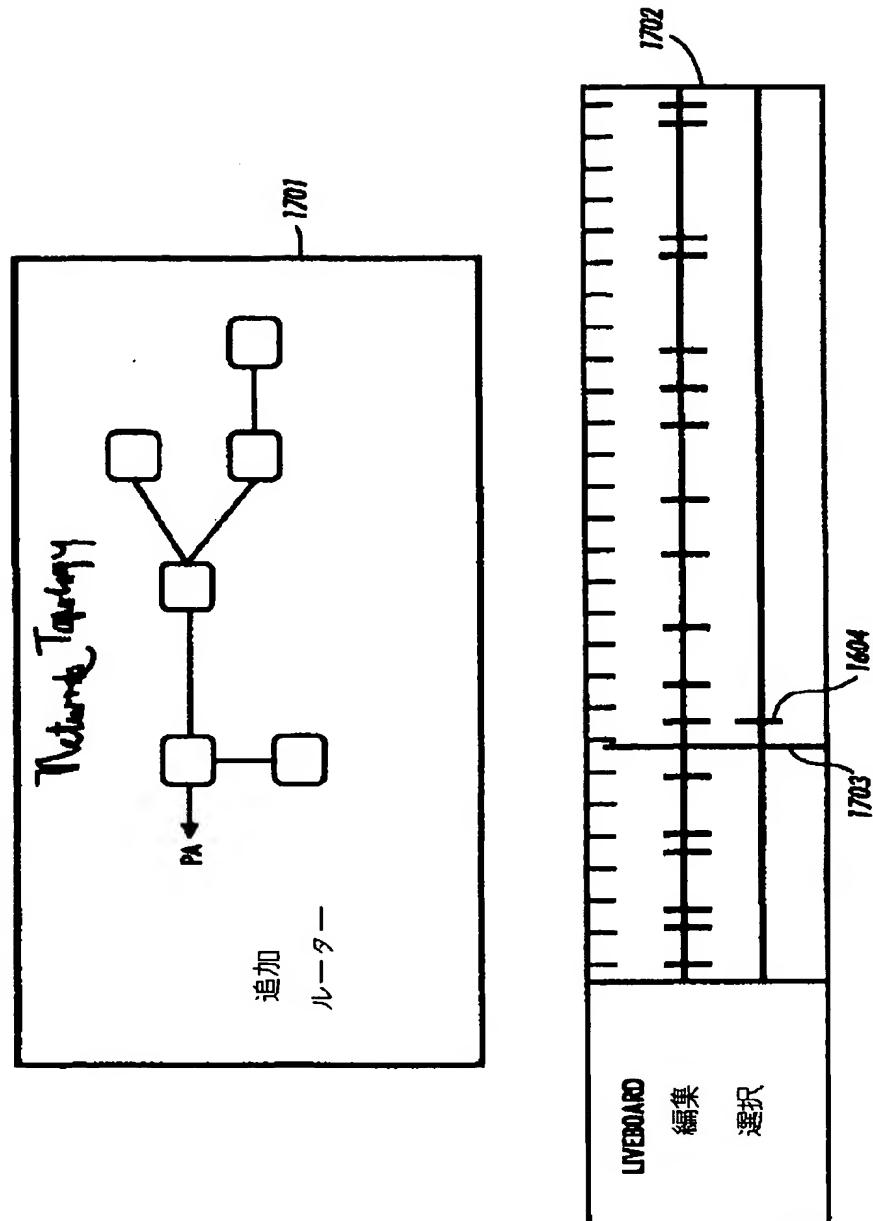
【図15】



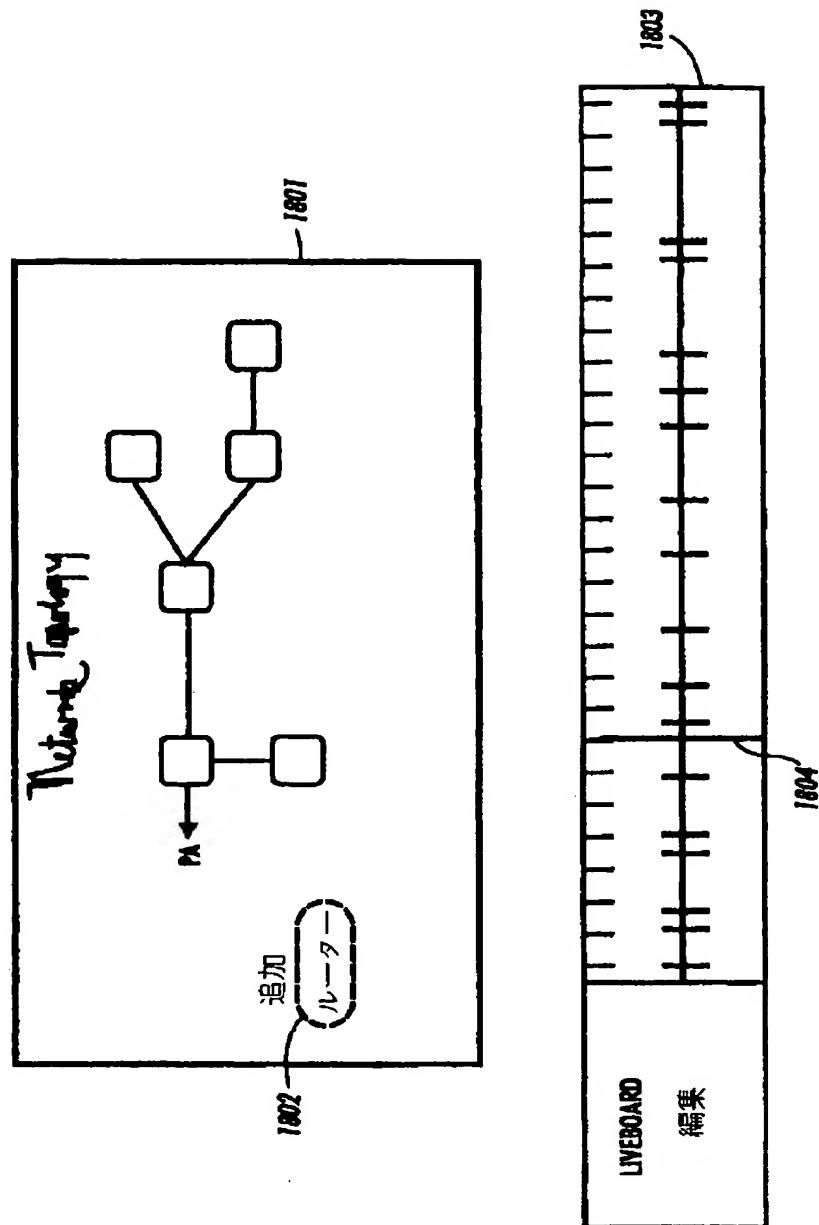
### 【图16】



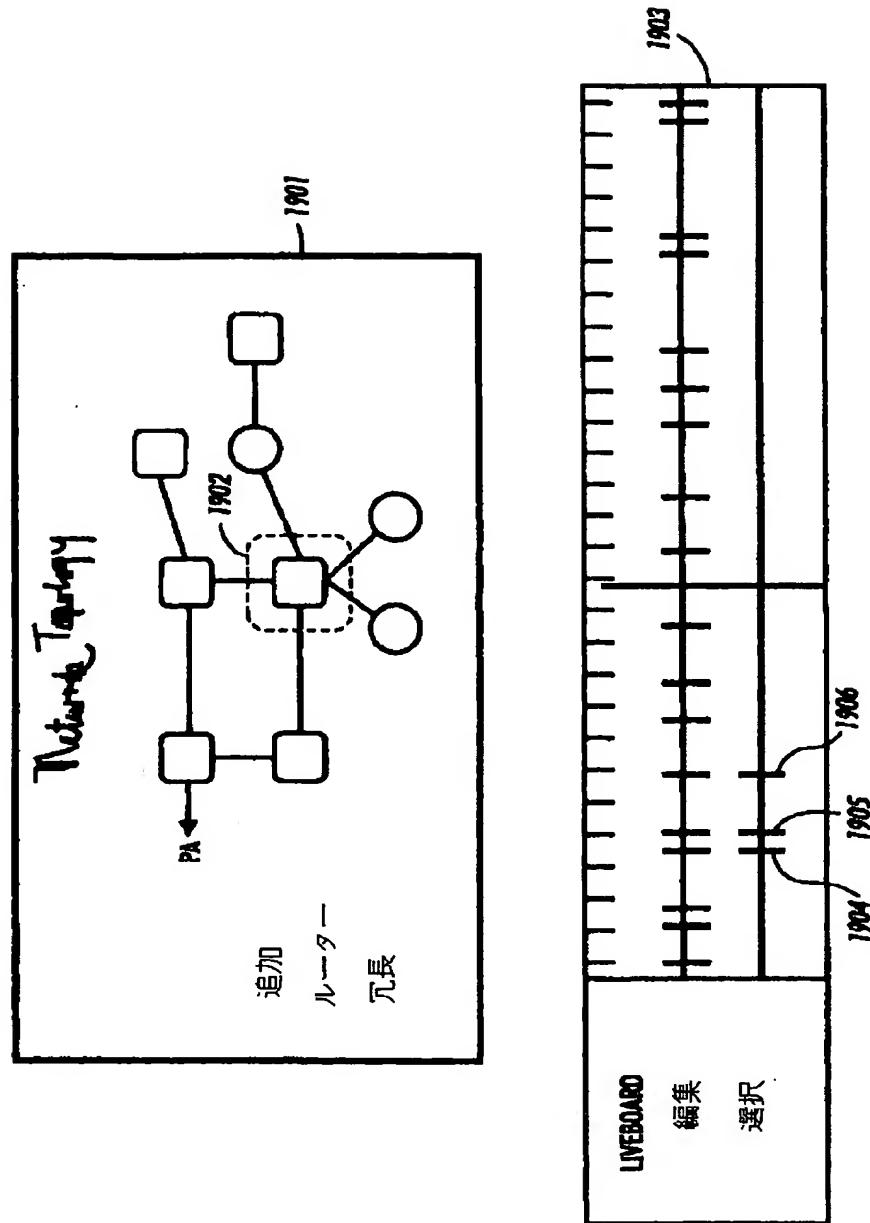
【図17】



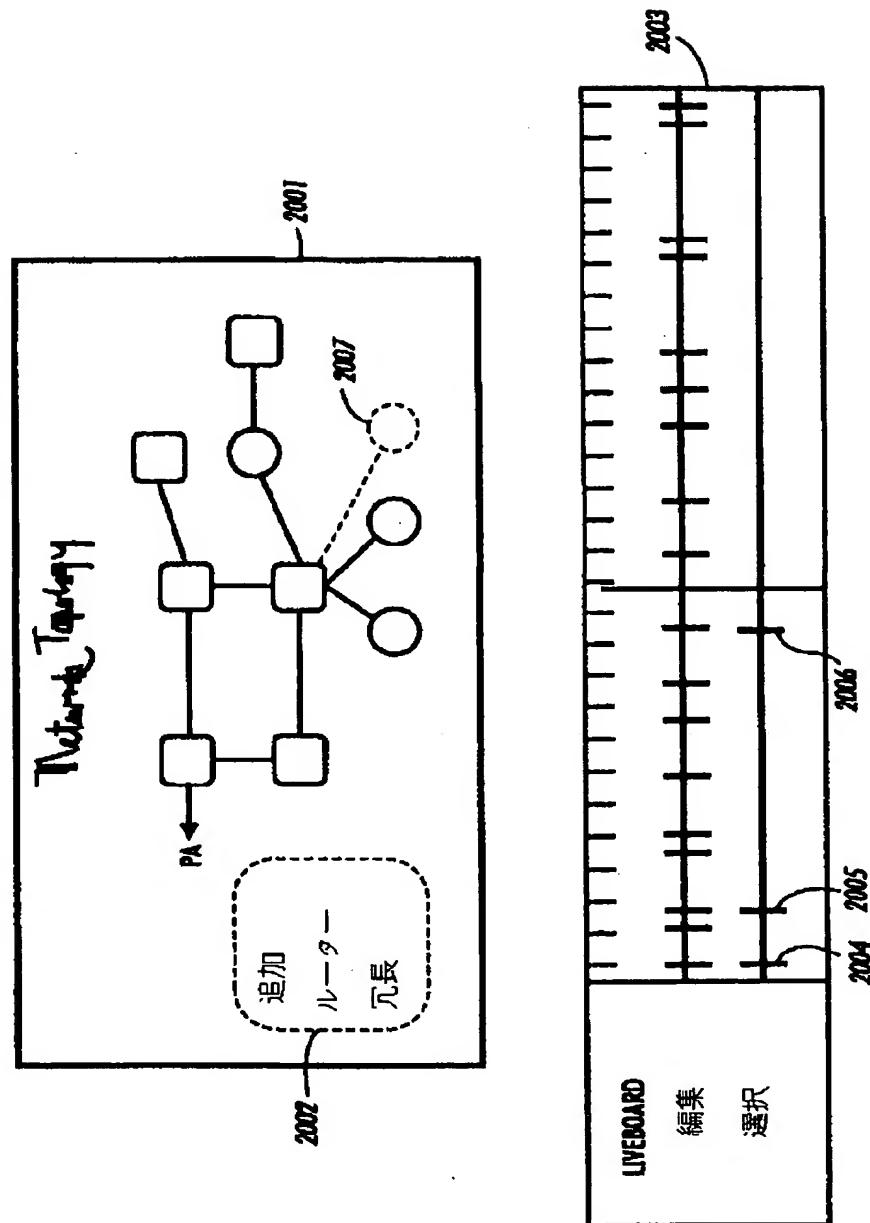
【図18】



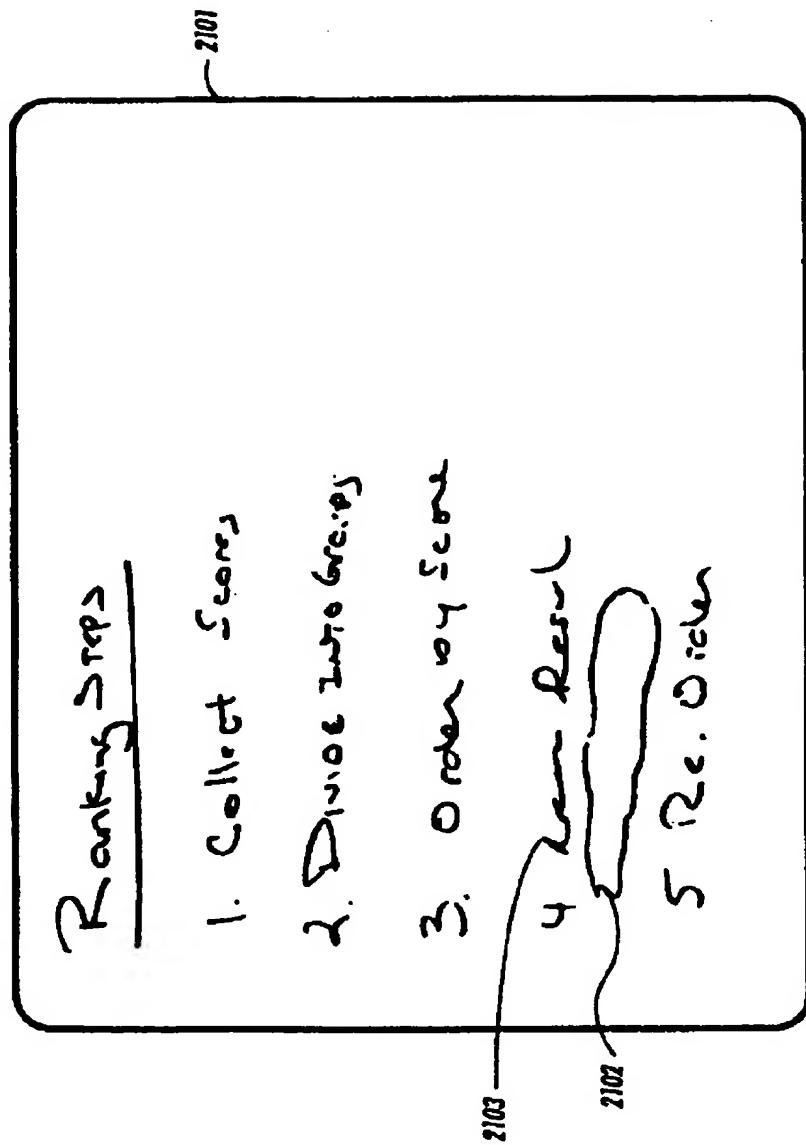
【図19】



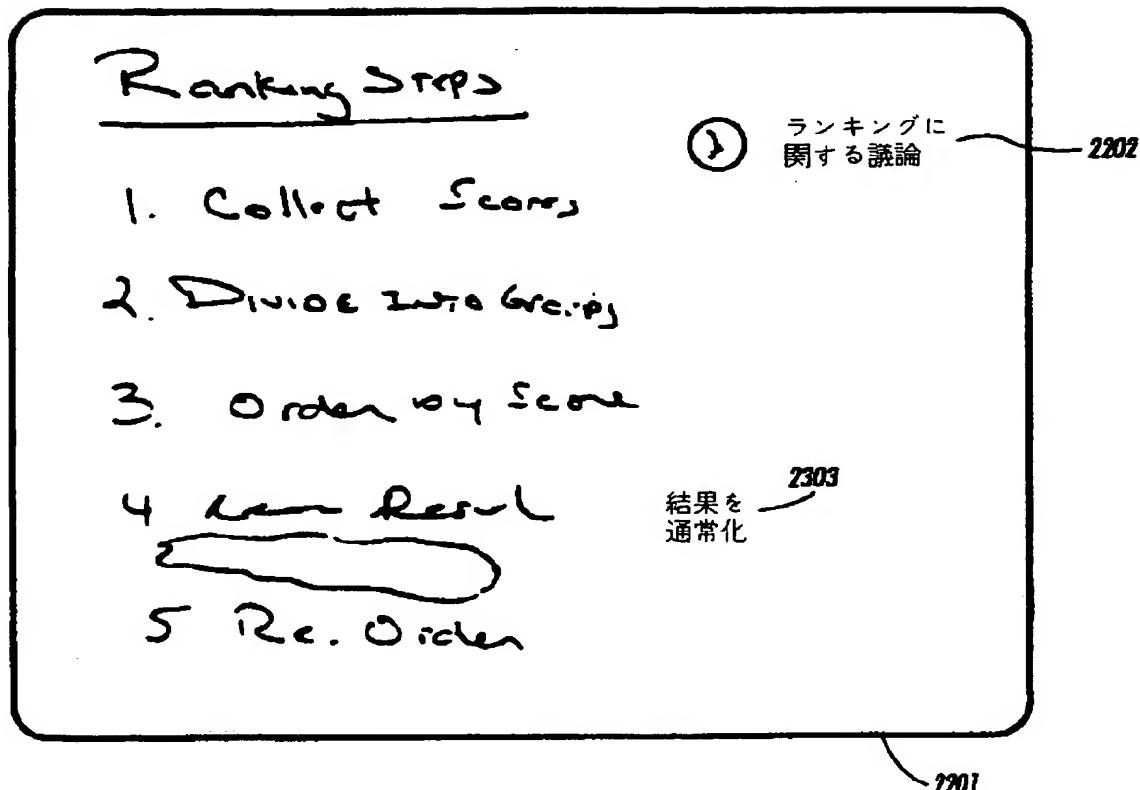
【図20】



【図21】



【図22】



フロントページの続き

(72)発明者 ドナルド・ジー・キンバー  
 アメリカ合衆国 カリフォルニア州  
 94040 マウンテンビュー ピクタースト  
 リート #3 678

(72)発明者 ウィリアム・ジェイ・バンーメッレ  
 アメリカ合衆国 カリフォルニア州  
 94022 ロスアルトス ディステルドライ  
 ブ 651

(72)発明者 ゴードン・ピー・クルテンバッハ  
 カナダ エムラエイ2ジェイ5 オンタリ  
 オ トロント スプルースストリート  
 126

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-171448

(43)Date of publication of application : 30.06.1997

(51)Int.CI.

G06F 3/14

(21)Application number : 08-288851

(71)Applicant : XEROX CORP

(22)Date of filing : 30.10.1996

(72)Inventor : THOMAS P MORAND

DONALD G KIMBER

WILLIAM J VAN-MELLE

GORDON P KURTENBACH

(30)Priority

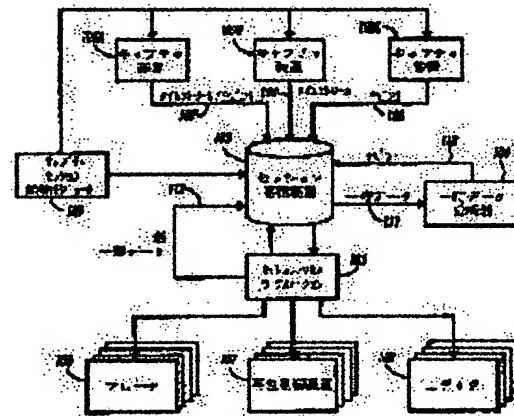
Priority number : 95 552663 Priority date : 03.11.1995 Priority country : US

## (54) COMPUTER CONTROL TYPE DISPLAY SYSTEM

### (57)Abstract:

**PROBLEM TO BE SOLVED:** To control the reproduction of temporary data expressing a joint work such as a meeting by detecting a user mutual action by either one of a first and second window, making each of a shared expression media emulation circuit and a time line generation circuit generate a control signal and updating the first and second window according to the user mutual action.

**SOLUTION:** A capture session starting module 101 synchronizes the starting of a 'recording' by each capture device 102 A to C in the same way. Each capture device is capable of starting the recording simultaneously or at different time. Each of the capture device 102 A to C generates an event in the instance in which the time streams of various kinds of media can be acquired. The generated temporary data is stored in a session storage device 103. The access to a recorded session is attained by a session access station 105.



### LEGAL STATUS

[Date of request for examination] 30.10.2003

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-171448  
 (43)Date of publication of application : 30.06.1997

(51)Int.CI. G06F 3/14

(21)Application number : 08-288851 (71)Applicant : XEROX CORP  
 (22)Date of filing : 30.10.1996 (72)Inventor : THOMAS P MORAND  
 DONALD G KIMBER  
 WILLIAM J VAN-MELLE  
 GORDON P KURTENBACH

(30)Priority

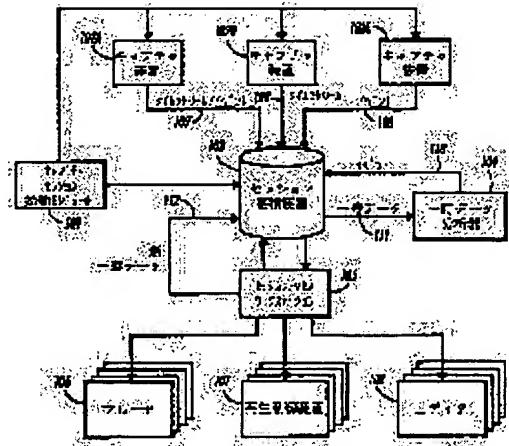
Priority number : 95 552663 Priority date : 03.11.1995 Priority country : US

## (54) COMPUTER CONTROL TYPE DISPLAY SYSTEM

### (57)Abstract:

**PROBLEM TO BE SOLVED:** To control the reproduction of temporary data expressing a joint work such as a meeting by detecting a user mutual action by either one of a first and second window, making each of a shared expression media emulation circuit and a time line generation circuit generate a control signal and updating the first and second window according to the user mutual action.

**SOLUTION:** A capture session starting module 101 synchronizes the starting of a 'recording' by each capture device 102 A to C in the same way. Each capture device is capable of starting the recording simultaneously or at different time. Each of the capture device 102 A to C generates an event in the instance in which the time streams of various kinds of media can be acquired. The generated temporary data is stored in a session storage device 103. The access to a recorded session is attained by a session access station 105.



## LEGAL STATUS

[Date of request for examination] 30.10.2003

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

**\* NOTICES \***

**Japan Patent Office is not responsible for any damages caused by the use of this translation.**

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

---

**CLAIMS****[Claim(s)]**

[Claim 1] A computer control mold display system for controlling playback of a session gained with two or more media capture equipments characterized by providing the following At least one of said two or more of the media capture equipments is share expression media, said share expression media generate and display a graphic object, it connects with one or more players for said computer control mold display system to reproduce said session, and said computer control mold display system displays. A cursor controller for enabling a user to act on mutual [ said / computer control mold display system and mutual ] That to which it is a session input means for receiving temporary data of said gained session, and said temporary data changes from a time stream and an event Are the processing circuit connected with said session input means, and it consists of what generates display information for a window where said processing circuit was displayed on said display. Said processing circuit is a share expression media emulation circuit for generating display information from a time stream expressing an interaction with said share expression media for a display of the 1st window of said display. Said 1st window As a player for displaying temporary sequence of marking visually by said share expression media Moreover, a thing which functions as a user interface for controlling playback of said session, It is a time-line generating circuit for generating display information for displaying a visual indicator on time line of the 2nd window of said display. What each of said visual indicator expresses a temporary index to said session to corresponding to an event, A user interaction of either said 1st window or said 2nd window is detected. A user interface control circuit which is made to generate a control signal over each of said share expression media emulation circuit and said time-line generating circuit, and updates said the 1st window and said 2nd window according to said user's interaction

---

[Translation done.]

## \* NOTICES \*

**Japan Patent Office is not responsible for any damages caused by the use of this translation.**

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

---

## DETAILED DESCRIPTION

---

### [Detailed Description of the Invention]

[0001]

[The technical field to which invention belongs] This invention relates to the field of the multimedia capture of a session, playback, and edit.

[0002]

[Description of the Prior Art] the meeting by which most groups' work was structured -- even when arranged setting [ like ] is performed, you can understand in common the modality in which the gestalt of a casual interaction like an exchange of the language by conversation is taken, and a group has the technology of work of them, a process, and relation by this. The focus of this casual activity suits to the result of such an activity, and it is not fully supported depending on the almost disregarded tool of computer use about how the group reached those results. Furthermore, the attempt which collects such information has resulted in making a participant follow a convenient method at an information gathering tool considering the activity as a formal thing rather than it supports the practice of natural work of them.

[0003] Collecting the audio of a meeting, video, and computer records offers the record in which sufficient examination of a group process is possible. Though it was regrettable, the advantage of such record could not tend to treat easily because of often needed sequential access. It is possible to access such information immediately with such the advent of informational digital storage. However, immediate access is useful only when the location which should go is known. Therefore, it is required and desirable to attach an index to such record based on an element with semantics or an event.

[0004] One approach of attaching an index is called real-time note taking (Real-Time notetaking). And people use the system which carries out a time stamp to each various elements of a note, and take [ then, ] notes between meetings. This offers the index in which an understanding to record of a meeting is possible. By only choosing the element of the arbitration of a note, a system can search the portion of AV record corresponding to the time stamp of a note element, and can be reproduced.

[0005] The easiest example is the copy system of a sentence, and a user only types a text there as a meeting progresses (a text can be considered as the note of the arbitration about a literal copy or literal meeting of having spoken). A note consists of the string of a text and the note element by which the time stamp was carried out is the discrete character of a text. After note taking is completed, a user can choose an alphabetic character, and the portion to which meeting record corresponds will be searched and reproduced.

[0006] A note taking system enables a user to constitute a visual expression (namely, note). And the visible element achieves a function as an index to meeting record. Each index element has the time amount related with it. This time amount is the time amount to which the user generated it. Since each index element is visible, a user can only choose it as it with directions, and since it has the associated time amount, the address can be obtained to meeting record.

[0007] A note taking system operates good, as long as the structure of a visual expression consists only of the sequence of the element which generates an event. If an expression is allowed to be edited between the courses of a meeting (i.e., if element modification and an element-deletion event exist similarly), a problem will arise. An index is a visible element and the cause of the problem is not being an event.

[0008]

[Problem(s) to be Solved by the Invention] The disadvantageous point of the approach by note taking is that people have to concentrate on work of note taking. Therefore, it is desirable to generate an index into meeting record without making people concentrate on work of note taking which is the natural side effect of meeting activity itself.

[0009]

[Means for Solving the Problem] This invention aims at a user interface for controlling playback of temporary data expressing a joint activity like a meeting. Such temporary data is gained by one or the capture equipment beyond it (for example, record or an electronic feltboard of an audio). Temporary data consists of a time stream and an event. A time stream is data with which the time stamp of the set expressing an aspect of affairs in which a certain record of a meeting is possible was carried out. Record of an audio and video and hysteresis in which action on an electronic feltboard was accumulated are the examples of a time stream. An event is an occurrence between time streams. An event is used in order to generate an index which offers direct access to one point or a span between joint activities. A time stream may be analyzed in order to define an event as a proper or to identify an event. Event information consists of the various properties of a time stamp, an event mold, and an instance of an event.

[0010] Playback of a session is performed under control of session access equipment. Two or more players for reproducing a time stream were connected with session access equipment. Session access equipment uses event information, in order to generate a user interface for controlling session playback. A user interface consists of two or more windows. Each window can express a player, a playback control unit, or an editor (as for this, a user permits generating time stream information on additional). Two windows are used in this invention by the coherent way of carrying out a having-two-incomes operation. The 1st window achieves a function as a player for reproducing a time stream, and an electronic feltboard is used in order to control playback of a session similarly. The 2nd window achieves a function as a playback control unit using a time-line metaphor. In the 1st window, each marking after an electronic feltboard expression is related with one or an event beyond it (for example, event generation, event edit, event migration, or event deletion). Selection of marking makes it possible to be moved to one time amount of an event with which playback time amount was related. When chosen, an event is displayed on time line of the 2nd window. On the contrary, when a visual indicator of a time-line window is chosen, an object to which the 1st window corresponds is highlighted. Such coherence enables a user to set up an aim in a point in a session which probably includes interesting information correctly.

[0011]

[Embodiment of the Invention] Drawing 1 is the block diagram of the system for acquisition of temporary data in this time of this invention, and a suitable embodiment, and playback. Drawing 2 is the block diagram of the session access workstation which can be used in this time of this invention, and a suitable embodiment. Drawing 3 is in the false condition that actuation of the session access workstation of this time of this invention and a suitable embodiment was simplified. Drawing 4 is drawing of the visual user interface of the display connected with the session access workstation of this time of this invention, and a suitable embodiment. Drawing 5 is drawing of the time-line user interface display for the session of this time of this invention, and a suitable embodiment. Drawing 6 is drawing where the time-line interface which highlights the relation between general-view time amount truck display area and focal time amount truck display area was simplified. Drawing 7 is drawing showing the result of the actuation into which the focal bar of drawing 6 is extended and the size of focal time amount truck display area is changed according to this. Drawing 8 is drawing of the time-line user interface display system for controlling playback of two sessions. Drawing 9 is drawing of the time-line interface with which two discontinuous focal segments are displayed on general-view time amount truck area, and focal time amount truck area includes the details of each focal segment. Drawing 10 is drawing of the time-line interface in which the result of size correction of the focal time amount truck area of drawing 9 is shown. Drawing 11 is the example of the time-line interface for playback of the session recorded with two or more capture equipments including the truck section produced as a result of selection actuation of a LiveBoard window. Drawing 12 is the further drawing of the time-line user interface display in which the result that the truck of the audio of drawing 11 was merged on one truck is shown. Drawing 13 is the further drawing of the time-line user interface display of drawing 11 which shows an open beam result for a note on a note truck. Drawing 14 is drawing of the meeting player about the same session used for generating the time-line user interface of drawing 11. Drawing 15 is drawing of the LiveBoard window in this time of this invention, and a suitable embodiment. Drawing 16 is drawing of the functionality between a LiveBoard window and a time-line interface. That is, it is index attachment by the time line from a graphic object. Drawing 17 is drawing of the functionality between a Liveboard window and a time-line interface. Namely, both a LiveBoard window and a time-line interface are in the synchronized display condition reflecting the same thing as coincidence. Drawing 18 is drawing of the functionality between a Liveboard window and a time-line interface. That is, it is index attachment by that of space from a temporary thing. Drawing 19 is drawing of the functionality between a LiveBoard window and a time-line interface. That is, they are many events per graphic object. Drawing 20 is a LiveBoard window and the chart of the functionality between time-line interfaces. That is, an index is attached by the location of object area. Drawing 21 and drawing 22 offer the example of the LiveBoard window in a "BAUNSHINGU-ball" playback mode before and after editing a LiveBoard window.

[0012] playback of record of a joint activity [ like a meeting ] whose this invention is, and other together non-working activities -- providing .

[0013] The purpose of this invention is generating an index to meeting record of the joint activity which is the natural by-product of the activity itself. This offers the intuitive method of obtaining a part of direct access of the session when people were recorded. An event like taking modification of a speaker, the writing to a feltboard and marking actuation, and a note is the example of the natural by-product of the meeting which offers the useful index point.

[0014] The system of this invention is described with reference to the time stream architecture and system architecture based on software. Time stream architecture describes the software framework by which this time and a suitable embodiment are performed. System architecture describes the component of the function of this invention.

[0015] The term used by description of a time stream and system architecture is defined here.

[0016] A time stream (Timestream) means the media data which was gained with capture equipment and by which the time stamp was carried out, and can reproduce it with a player.

[0017] An event (Events) is a shift of the occurrence generated at intervals of a certain point between the activities gained, for example, a speaker, or the writing to a feltboard.

[0018] Temporary data (Temporal Data) means association of a time stream and an event.

[0019] The collection object of temporary data expressing the activity gained by sticking a session (Sessions) like a meeting is named.

[0020] A player (Players) is the gestalt of the origin of it, or is the gestalt which shows the dynamics of a time stream, and can reproduce a time stream.

[0021] An editor (Editors) can change a visual indicator or means the user interface which can add an event and a time stream to a session.

[0022] A playback control unit (Playback Controllers) means the user interface by which playback of a session is controlled.

[0023] Capture equipment (Capture Devices) is equipment which gains and stores temporary data. The capture equipment of a different mold gains the media (for example, writing on an audio, video, and a feltboard) from which the session differed.

[0024] Time stream architecture supports use of the tool of the harmonious record and a set with the expandability for playback. Although this exists in one address space of a specific machine, it is based on the distributed object system which enables a user to build easily the object which has the "substitute" object which exists in an address space which is the identitas of the identitas in a network, or other machines, or is different. This distributed object system is called the unification (ILU) between language (Inter-Language Unification), and is described in the "ILU manual" by Janssen, the Xerox technical report, and ISTL-CSA-94-01 January, 1994 [ -02 or ]. The fundamental data components of time stream architecture are an event, a time stream, and a session.

[0025] An event consists of a certain possible property for which it may depend on start time, a period (it is probably zero about a 'point' event), a mold, and a mold. (For example, a LiveBoard page event may have 'pagination' and a 'page title' property.) The interface to an event object is as follows.

TYPE Event=CLASSUPERCLASS PersistentObject ENDMETHODS  
 getType():String, getStartTime():Time,  
 getDuration():TimeEND  

[0026] Although the interface specification expressed here is carrying out the style of the assignment language of the ILU system mentioned above, it can be literally come out of it as it is, it cannot mean a certain thing, and can translate it easily by the person expert in object oriented programming. The semantics of a data type string and time amount is obvious. Additionally, an event is getPropertyNames():StringSequencesetProperty  
 (name:String, value:String) which inherits the following methods as a subclass of PersistentObject.

getProperty(name:String): String[0027] An event is generated by the call to a session object as stated below.

[0028] A time stream object is used in order to refer to the data stored for the predetermined media mold. Speaking generally, the data relevant to a time stream enabling access based on the time amount of the recorded media. like the recorded audio, many digital media are related and the translation of data which consists of the data time amount sample data which this was able to consider was an event, for which it depends on media in large quantities and greatly also although kicked (again -- equipment) and by which the sample was carried out makes it much more practical to regard a time stream as 'opaque'. That is, time stream data is operated by normal only with the special tool connected with the media of a time stream, and a format. Especially data is generated by the recorder object and reproduced by the player object. Similarly, in order to make an event, the specific tool of media can also analyze. (For example, a speaker segment, silence/interruption, scene change, etc.) By dealing with a time stream object as opaque, it should be cautious of the ability of architecture to be adapted for analog media. (For example, a recorder and a player may be the video tape recorders controlled by computer software, but a time stream may offer the software equipped with the information

for which the tape should be used, and the absolute time which should be related with the start of a tape.) The interface to a time stream object is as follows.

TYPE Timestream=CLASS SUPERCLASS PersistentObject END METHODS mediaType():String e.g. audio format():String e.g. sun au startTime():Time duration():Time END [0029] Sessions are an event, a time stream, and the collection object of the layered structure which may contain other sessions. It should be cautious of it being a wide sense, for example, a predetermined meeting or the 'session' of the narrow semantics of the data relevant to discussion being a special case from what this use of term 'session' sometimes encounters. A session is a general-purpose bookkeeping device for essentially organizing time stream data. The interface to a session object is as follows.

TYPE Session = CLASSSUPERCLASS PersistentObject ENDMETHODSgetTimestreamByName(name:String):Timestream, getAllTimestreams():Timestream Sequence, getSessionByName (name: Session): Session, getAllSessions ():Session Sequence, createEvent(start:Time,dur:Time, type:String, props:PropertyList:EventEND[0030] It is extensible in the semantics that the set of the mold of the media which time stream architecture was sharing the common interface with which the player and the recorder object have been independent of a media mold, and were supported is not being fixed. The simplified interface to a player and a recorder object is as follows.

TYPE Player=CLASS METHODS seek (t:Time) Playback time amount is changed into t. location():Time It returns to the newest playback time amount. play (second:Speed) It (1= usually) reproduces with a velocity coefficient s. stop() Playback is suspended. ...

attach (timestream:Timestream) A player is set in order to reproduce a predetermined time stream. END;

TYPE Recorder=CLASS METHODS record(), pause(), createRecording(session:Session,name:Name):Timestream, finishRecording(), END;

[0031] The player about a predetermined media mold and a recorder object are carried out by 'MediaServer' relevant to the media mold. Although supporting a new media mold naturally needs operation of new MediaServer, change of architecture (to namely, interface required whether a recorder and a player are made and in order to control) is not needed. This can write in general-purpose application (for example, a record control program or access, and a playback control program), and means that implementation of the new service about a new media mold or the existing media mold will be supported automatically. The description by which the interface to a MediaServer object was simplified is given by the bottom type.

TYPE MediaServer= CLASSMETHODSgetRecorder(format:String, context:String):Recorder, getPlayer (format:String, context:String) :P layerEND [0032] The argument of the context specifies the information about a player or a recorder. For example, the host and device name which were used for the audio input / output can be specified.

[0033] An analyzer processes the data relevant to one or the time stream beyond it, and it can be used for it in order to make an event. Such two analyzers used in the suitable embodiment at present are described below. The description by which the interface to an analyzer was simplified is given by the bottom formula.

TYPE Analyzer=CLASSMETHODSproduceEvents(session:Session,specs:String)

END [0034] An analyzer accesses the data of the time stream of a predetermined session, and generates the event relevant to a session. Which type of time streams (for example, an audio, video, etc.) are accessed by the analyzer depending on the mold of an analyzer. The argument on 'specs' offers all the required information about the method by which analysis is performed. For example, an analyzer can be used as the WORD monitor machine of the audio which generates an event whenever it speaks about predetermined WORD (they are some specs strings). The analyzer should be noticed about the ability to be made to operate on real time between record processes after a session is recorded. In the case of real time, the event made by the analyzer 'is observable' with other processes (refer to the notice of an event described below) like a process of carrying out a user interface.

[0035] In order to promote this expandability further, architecture offers the broker object which arranges suitable MediaServers for supporting the media of the specified mold. A broker can assist arranging the object of other classes like an analyzer similarly. A broker can maintain the truck which must be performed in order that an activation process may carry out a predetermined object mold (for example, MediaServer of an audio) and/or may generate these processes using the information accumulated in the shared network file. The name service of other network bases was able to be used similarly. The description by which the interface to a broker was simplified is Type given by the bottom formula.

Broker=CLASSMETHODSgetMediaServer(media:String):MediaServergetAnalyzer(type:String):AnalyzerEND[0036]

An event, a time stream, and session data are stored in a time stream data base. The application program using a time stream data base obtains first the time stream master object which has whether other objects are made and the method of arranging. A master object can be obtained from the time stream server performed in another process (on host another probably) which uses the same device as what was used by the broker mentioned above. Or the program code which

carries out a time stream server (and master object) can be directly included in an application process.

[0037] The fundamental step required for the easy record application by the pseudocode is as follows.

... 'Master' The top-level time stream object called is obtained.

```
session=master.getSessionByName('Recordings' )
```

```
broker=master.getBroker()
```

```
mserver=broker.getMediaServer('audio')
```

```
recorder=mserver.getRecorder('mu law', 'default audio device')
```

```
recorder.createRecording(session, 'recording1')
```

```
recorder.record()
```

... A certain code looks for the user interface event which shows what record should be a pause, a resume, or completion.

... The routine for closing record is called.

```
recording.finishRecording()
```

exit [0038] The above-mentioned pseudocode fragmentation describes the step needed in order to generate the audio record named 'recording1' of the existing session called record.

[0039] The pseudocode required in order to reproduce record is as follows.

... The top-level time stream object called a master is obtained.

```
session=master.getSessionByName('Recordings')
```

```
timestream=session.getTimestreamByName('recording1')
```

```
broker=master.getBroker()
```

```
mserver=broker.getMediaServer('audio')
```

```
player=mserver.getPlayer('mu law','default audio device') player.play(1)
```

It waits for the user interface event which shows the thing for which playback should be suspended, or the thing for which different retrieval of time amount should be called.

[0040] Time stream architecture assists the application which generally cooperates mutually, and offers especially some devices in which the cooperative control to many players and recorders is assisted. One device is a union (Confederation) object which is the collection object of the player and/or recorder which operate as a single player and/or recorders. A player or a recorder is dynamically added to union, or can be deleted from union. The description by which the interface to union was simplified is given by the bottom type.

TYPE Confederation=CLASSSUPERCLASSES Player Recorder ENDMETHODS registerPlayer (player:Player), unregisterPlayer (player:Player), registerRecorder (recorder:Recorder), unregisterRecorder (recorder:Recorder), and ...

END [0041] As a subclass of a player and a recorder, it is suggested that union supports the method of those classes like playback and record. Moreover, union also supports the additional method of acting as playback of the sequence of a time amount range (behaviors).

[0042] The code simplified for setting and controlling playback of all recorded data medium about the session called 'ProjectMeeting#3' using union is as follows.

Top-level time stream object called a master ... session=master.getSessionByName 'ProjectMeeting#3'

```
...broker=master.getBroker()
```

```
...conf=master.getConfederation('xyz')
```

..timestreams=session.getAllStreams() for each timestream in timestreams do begin ...mserver=broker.getMediaServer (timestream.mediaType())

```
...player=mserver.getPlayer(timestream.format(),'host1')
```

```
...conf.registerPlayer(player)
```

```
end conf.play()...
```

...

It waits for the user interface event which shows how playback is controlled.

[0043] It should be cautious of the identifier of 'xyz' being given to union. Other programs obtain this union from a master object, and although they control playback using this, they are made. Although this makes it possible to perform the program of some [ the same host top ], however in order to control playback jointly, probably it needs a very different interface. In addition, although it is the same as having mentioned above, the program performed on a different host can generate a player in relation to the host, and can register them with the 'xyz' configuration. A result serves as a playback tool of share distributed process input output equipment which does not need the programming burden of the addition for carrying out for an essential target.

[0044] Another adjustment device is the notice of an event. A program can carry out EventNotifier which is the object

which has the method called for 'noticeEvent'. A NOTI fire can be registered with a session object, and subsequently to the session, whenever an event is added, it is then called with an event as an argument in an event. For example, in a LiveBoard window, when some objects are chosen, the event which shows the time amount relevant to the selected object can be generated, the time-line interface equipped with registered Eventnotifier can be contacted (namely, the called noticeEvent method), and such time amount can be shown in graphic. This example is stated to details below.

[0045] The problems of a synchronization are various joins and are generated with various gestalten. There is the need of saying that a time stream is changed into the condition of having fully aligned as it is not made to become irritated with the beginning like between perception of playback of an error between captures. In the case of an audio and video, this means that video and a sound must continue making it have to align in about 1 / 30 seconds (for example, before a problem lip synchronous [ irritating ] occurs). Since two time streams may be gained independently on a different machine, the thing using a certain common clock (or it rebuilds) is required for them among various kinds of sources. At present, in a suitable embodiment, while performing by connecting completely The network time amount protocol (NTP) stated by RFC-1769 In order to maintain a clock within a ms at a precision, it is used between distributed machines (this is the protocol of an ecad). By only carrying out resynchronization to a common standard, it is not and a result better than usual is obtained by learning how the drift of a machine is suited. Suitably, although the time stump of each amount of the data of a time stream is carried out, the exact clock rate of the certainty of the system that a sample is carried out is enough as data similarly (since the real-time protocol packet was used by video).

[0046] In other cases, in a comment of the sentence which attaches an index to other media time streams, the demand for such a strict synchronization can be eased. The metrics for which any errors needed again sufficient accuracy made not to come to become irritated between playbacks are applied. About the note of a sentence, a note means that an index must be attached to a time stream in the location enough near what the user meant so that this may be a functional pointer in the material with which the mark was recorded. the bottom of a certain condition -- several seconds -- falling -- although it is permissible, in other cases, even the fraction for 1 second may be nonpermissible If the user is running NTP on the hardware which is running comment software, generally as for the measurement, sufficient accuracy will be offered. Other options include synchronizing a clock with initiation of a meeting, in order to call a time amount server in order to acquire time amount current whenever a comment is carried out from a NTP equipment machine, or to calculate a comment machine, that of NTP time amount, and offset of a between. Correction of the gestalt of this last may use the gestalt of a special "synchronous event." It comments on this in the data beyond two or it of a distributed process input output equipment resource, or it can be analyzed from it.

[0047] It is ready for operating almost regardless of the degree of the synchronous accuracy demanded, where various capture equipments are separated completely. This is desirable [ connection (a computer an audio, video, etc.) is not desirable, or ], when record is performed in impossible setting. If the means for finally aligning the clock of equipment of a different kind is offered, a hand held computer or a commercial camcorder is acceptable capture equipment. The example of the technology which aligns the separated laptop at record and the single tier of an audio is striking a laptop key loudly, and is an effective synchronous event. A keystroke is taken up by laptop, a time stump is carried out with that clock, an audio is analyzed in order to find out the audio event expressing a key hit, and the time stump of this event is carried out with the clock of an audio recorder. From now on, offset between two clocks is calculable. And one side of the set of a time stump doubles with another side.

[0048] A drift (very low frequency error) is a problem similarly during the session to record. In order to amend systematic gain or loss depending on the degree of the accuracy needed in the session when the independent clock was adopted, or in order to carry out the repeat check of the clock for offset amendment, it may be necessity to read initiation and termination of a session. If data is collectable in order to propose corrective actuation ( 8057/8000 when [ for example , ] speed playback is usually a request playback ), the drift in these elements of the system ( for example , audio equipment it was presupposed that was operated in 8000 samples / second although 8057 samples / second should come ) in which these selves carry out clock actuation may need additional fine tuning between playbacks .

[0049] A jitter (still higher frequency error) is processed by generally carrying out a buffer in the suitable location of a system.

[0050] The problem synchronous [ in a time in-stream playback session (it may often generate in coincidence with a capture session, or may be in the "nest condition" of the kind set to setting of a capture) ] is approaching further what is typically dealt with in a multi-media system.

[0051] In order to set the context, it is assumed that it is accumulated while synchronizing closely and reproducing all of these time streams, or a desired subset so that various kinds of media time streams (an audio, video, count log) may be gained and the gained activity can re-experience. Furthermore, record is altogether equipped with perfect data and it is assumed that it was gained in the drift which should be amended, the jitter, or the condition that there is no offset.

[0052] At present, in a suitable embodiment, in order for various time stream players to approach moderately and to synchronize them, the non-drift system clock which fully synchronized is used for them, and they operate in freewheel mode. Each player is a method suitable for progress of a system time, and this means that there is responsibility which reproduces the time stream. In an ordinary playback speed, 1 second of playback (and system time progress) means that it should generate correctly for every second of original disk record, and this only expresses that there is much more much still quicker record by which a still later playback rate is reproduced by reaching, or there is. [ still less ] It reaches, the best expression of low speed playback is noticed about these high things customizable if needed for a user, and the video clip of the quadrant second which it may be wished in order that a high-speed pitch conservation speech may know a meeting roughly, or carries out a chain from a suitable gap may most often tell high-speed video recovery very much.

[0053] Those with a thing and it to which two aspects of affairs of using a freewheel synchronization deserve describing are the opportunity of good initiation and a pause. The corrective action of such an approach is only dependent on the player which all start to coincidence regardless of [ the 1st thing ] a transmission time, buffering, etc. It means that this should receive the request about initiation by which the player was planned (it is X system time (although it is few, playback is started to sufficient time amount (fraction of a second) to the future) at 100% of usual speed of the material recorded for example, by Y system time.). This mode of initiation guarantees the harmonious initiation to which the real time clock tenacity (Adherence) following a degree produces desired actuation. User action of specification [ the opportunity of a pause ] only is part synchronous amendment (for example, one player which should be a master is declared, and whenever a user suspends or stops playback, he synchronizes all the Childe players with it.) to a player.

[0054] The block diagram of the hardware architecture of this invention is explained with reference to drawing 1. If drawing 1 is referred to, the session starting module 101 will perform various "housekeeping" functions as preparation which records a session. A session starting module is suitably materialized as software performed on the system based on the computer connected with various record components. Between the task, a session starting module must confirm having sufficient storage capacitance for storing temporary data which can generate session are recording equipment 103 which can be used. This is attained by performing presumption about the length of the session recorded, and several/mold of capture equipment used. The capture session starting module 101 synchronizes initiation of "record" similarly by each capture equipment 102 a-c. Each capture equipment can start record to different time amount simultaneous.

[0055] Capture equipment 102 a-c gains the time stream of various media, and generates an event in a certain instance. Capture equipment 102a generates a time stream and the event data 107. In an embodiment suitable at present, the example of capture equipment 102a is an electronic feltboard like the product LiveBoard (trademark) of LiveWorks (trademark) of California and Xerox Company (Xerox Company) of San Jose to available marketing. The time stream data gained from LiveBoard is generated in asynchronous as an interaction occurs. An event may be similarly generated by the call of a predetermined function (it is modification of a page on LiveBoard).

[0056] Capture equipment 102b generates the time stream 108. In the capture equipment of such a mold, capture equipment is in the mode in which time stream data is gained continuously. The typical examples of capture equipment 102b are video and the recorder of an audio. Another example of such capture equipment 102b is a pointer. Pointer capture equipment gains a motion of the laser pointer typically used between presentations, in order to point to something on a pointer, for example, the projected slide, or other vision members. Pointer capture equipment is useful especially because of the presentation of a positive type, when the material generally seen is prepared beforehand. Playback of a pointer time stream shows a motion of the pointer between playbacks of a presentation in this way.

[0057] Capture equipment 102c only generates an event 109. The example of capture equipment 102c is a carbon button. A carbon button is equipment which relates generation of the event of a carbon button time stream with the user who operates a carbon button. A carbon button event may be generated by the user in order to point to initiation of an interested topic, an especially useful dialogue, and the various activities [ like ] from which a topic switches. Another example is a slide projector. The event in a slide projector time stream points to modification of a slide.

[0058] Capture equipment 102 a-b gains a time stream in a digital format suitably. In order to make easy direct and random access to the various portions of a time stream, it is desirable to accumulate a time stream in a digital format. However, it is within the limits of this invention to gain and store time stream data in an analog format. The format of a time stream corresponds to the capture equipment used in order to generate a time stream. For example, although the time stream of an audio consists of the sample of the audio of the lot taken at a certain regular gap, on the other hand, it is the history file of the primitive actuation (explained to the following details) to which the time stump of the LiveBoard time stream was carried out. management of the gained time stream data is what kind of method -- an imitation -- it is the method of regarding it as their being the most suitable, and is carried out by the various media

servers which can be chosen in order to store those data.

[0059] Generated temporary data is stored in session are recording equipment 103. Session are recording equipment 103 offers are recording equipment eternal about the mold with which data differed, i.e., session data, event data, and time stream data. In fact, different physical are recording equipment distributes and session are recording equipment 103 is managed with the component with which time stream architecture (it states below) differed.

[0060] Similarly, a temporary data analyzer 104 is connected with session are recording equipment 103. A temporary data analyzer 104 is a processing means (for example, system based on a computer) used in order to analyze temporary data 111, in order to identify an event 110. Event information is accumulated like session are recording equipment 103.

[0061] Access to the recorded session is attained by the session access workstation 105. This access can be made into direct continuation through a network (with for example, client-server architecture). The access session starting module 109 performs various "housekeeping" functions required in order to reproduce a session. Functionality of an access session starting module is typically performed by the session access workstation 105. The session access workstation 105 is a computer control mold display system typically, and the control to session playback is attained by the graphic user interface displayed on the display there. Such a graphic user interface is described below. Setting in the suitable embodiment at present, a session access workstation is SUN of California and Mountain View. Microsystems Sun equipped with the available X-Windows graphic user interface from Inc. OS SUN by which 4.1 operating systems are performed It is SparcStation-10.

[0062] Furthermore, the player 106, the playback control unit 107, and the editor 108 are connected with the session access workstation 105. Various players, a playback control unit, and each of an editor are unified in a session access workstation. Then, playback of an electronic feltboard time stream is attained by the 1st window on the display of a workstation, a video time stream is attained by the 2nd window on the display of a workstation, and an audio time stream is attained by the audio subsystem of a workstation. However, it will be clear to this contractor that a player does not need to be unified in a session access workstation and it can consider as the regenerative apparatus according to individual. Furthermore, in a suitable embodiment, a single window can perform a player, a playback control unit, and the function of an editor at present. The functionality which accompanies the organization of a window and them in a suitable embodiment at present is stated to details below with reference to drawing 3.

[0063] Finally, new temporary data 112 generated by use of an editor is stored in session are recording equipment 103.

[0064] The main functions of LiveBoard in a joint activity are offering "a share expression" which each user's can see and can act mutually. LiveBoard and its OPERETINNGU software should be noticed about being designed as such "a share expression" does not need to be on single LiveBoard equipment. Much LiveBoards(es) can connect together via a network so that it may be carried out in the physical location where actual joint activities differed.

[0065] an operative condition suitable at present -- LiveBoard which was used, having set like is a means for mainly generating an index to session record -- as -- it does not have intention. However, LiveBoard has the peculiar capacity used as the useful tool for controlling the playback next to the session which had it recorded so that details may argue further below. LiveBoard operates under control of Tivoli application. Tivoli application Pedersen, E., KMcCall, T. It Moran(s) and reaches. F. ""Tivoli:An by Halasz Electronic Whiteboard for Informal Workgroup Meetings" and Proceedings of the INTERCHI'93 Conferenceon Human Factors in Computing Systems, pp.391-389, April It is explained to 1993 at details.

[0066] Tivoli has the user interface similar to the class of a "drawing" program and the image generator known. A paint program generates and operates an image like the collection object of a graphic object. A Tivoli drawing is drawn as a graphic object. A graphic object contains the following.

- Stroke : "digital ink" OBUJIEKU generated with the pen input unit.
- Alphabetic character : it was generated via the keyboard or was carried in from the text file.
- An image / icon object : it was copied or it was carried in.
- Structuring object : it was carried in from the data base.
- Clock : it is generated with clock gesture. A clock is a special mark generated on LiveBoard which appears as a round analog clock dial face showing between the generate time.

[0067] Tivoli maintains the hysteresis list of performed primitive actuation. For example, the object which adds an object, which generates the object which deletes an object and which changes an object is chosen, or a page is changed. The time stamp of all the primitive actuation of a hysteresis list is carried out, and it is related with an object. This hysteresis list is analyzed in order to identify an event so that it may be stated to details below.

[0068] The graphic object which appears in LiveBoard is a method similar to whether it is made between the courses of a meeting, and actuation of opening the word processing file on a personal computer, and can be made into either of whether it is loaded from a certain memory data medium (for example, floppy disk). The instance of this latter is

especially useful when a session is a presentation of a positive type.

[0069] Another, important aspect of affairs of actuation of LiveBoard is use of gesture. Although gesture is generated like "an ink stroke (inkstroke)" of other arbitration, it is an interaction with LiveBoard interpreted as a command (for example, editing command). For example, if the specific object "should be deleted", an object will be chosen with the gesture surrounded with a pen, and the called Delete command (for example, pigtail gesture). After deletion arises, when a session is reproduced in this invention so that it may be stated later while marking caused by gesture only already makes it LiveBoard not to appear, a "ghost" image may remain.

[0070] As mentioned above, session are recording equipment is the functional component of the time stream architecture which offers the eternal are recording about the data, i.e., the session data, event data, and time stream data of a different mold. In fact, different physical are recording equipment distributes and session are recording equipment is managed with the component with which time stream architecture differed. It is most practical to divide session are recording equipment into a time stream data base (a session and for event data) and bulk time stream are recording equipment (for time stream data).

[0071] The actual data of a time stream is stored and bulk time stream are recording equipment is just going to be managed. In the case of video, this consists of the bulky sequence with the long video frame compressed [ was digitized and ] and digitized. time stream architecture -- setting -- management of time stream data -- what kind of method -- be -- the various media servers which can be chosen in order to store those data by the method of regarding it as their being the most suitable are left. At present, in the suitable embodiment, although most media servers stored those data in the usual Unix file, the media server for other are recording devices (for example, the computer control mold VCR for analog are recording of an audio / video time stream) was carried out similarly. About a media server, it is also possible to use the time stream data base for the time stream data which can be made into the most expedient device for the time stream equipped with intermittent few capacity data (for example, note of the sentence made by the laptop computer).

[0072] A time stream data base is an are recording location for storing and searching the data which systematizes a session and event data, i.e., all information, at a session, and describes the element of a session. A user positions a desired session, and a time stream data base determines which time stream is related with the session, and supports finding out an eclipse \*\*\*\*\* event with a ream to a session and its time stream. A time stream data base not only supports perusal of session data, but should be a data base which has question capacity for alternative retrieval of a session and an event as it is possible. It must continue over several months and several years, therefore a session and event information must be accumulated in non-volatile equipment (usually disk). Since time stream architecture is object-oriented, an object-oriented data base is suitable for it as object are recording equipment with the durability for a time stream data base. In a suitable embodiment, a time stream data base is carried out on a Unix file system at present.

[0073] Temporary date analysis is a specific thing to the information acquired by the event of the time stream which should be extracted, and a request. It is getting to know when the specific man speaking about one useful gestalt of an event in the case of the time stream of the information on an audio. Therefore, it is desirable to segment a time stream, since it means when a man different one is talking, to assign various segments to those who are talking two, and to analyze an audio stream. Therefore, these segments will express the event in a time stream. The technology for performing such analysis on an audio stream "Unsupervised by which both are transferred to the same grantee as this application Speaker Clustering For Automatic Speaker Indexing Of Recorded Audio The application number 08/No. 226,525 under coincidence continuation entitled Data", It reaches. "SegmentationOf Audio Data For Indexing Of Conversational SpeechFor Real-Time Or Postprocessing It is explained to the application number 8/No. 226,519 entitled Applications." Once such analysis is performed in any case, the list of events for the time stream of an audio will consist of two or more data items in which each contains a time stump, a period, and a speaker identification child.

[0074] In analysis of the time stream gained by LiveBoard, an event can be considered as the input to elimination of drawing of a drawing or the writing of language, page modification, a drawing, or language, the notes made to the existing marking, or the LiveBoard user interface of a specific command. As mentioned above, LiveBoard operates under control of Tivoli application. The time stream generated by Tivoli is a hysteresis list of primitive actuation which was recorded in temporary sequence and by which the time stump was carried out. Analysis of such a time stream is accompanied by analyzing the syntax of the sequence of primitive actuation for identifying a meaningful event.

[0075] Each marking on LiveBoard is called an object. Each object has one or the event beyond it related with it. It is required to identify all the events connected with the object and to be provided for a user as an index to the inside of a session for this invention. It is required to be checked by the user and to be provided for him as an index to the inside of a session. Therefore, temporary date analysis in this invention identifies the various events on an object, and accumulates the event. An object generation event exists about all objects. An object can have an object modification event and an object deletion event similarly. It argues about the further aspect of affairs of a LiveBoard time stream

below about a LiveBoard window.

[0076] At present, in a suitable embodiment, a session access workstation is a computer control mold display system which operates under control of the program instruction accumulated on a recording data medium, and is described with reference to drawing 2. When drawing 2 is referred to, the system based on a computer consists of two or more components connected by bus 201 course. The bus 201 explained here is simplified in order not to make this invention not clear. It can constitute from two or more parallel buses (for example, an address bus, a data bus, and a status bus) like the layered structure of a bus 201 (for example, a processor bus, a local bus, and an I/O bus). In any case, the computer system contains further the processor 202 for performing the instruction offered by bus 201 course from an internal memory 203 (the internal memory 203 should be typically noticed about it being the combination of a random access memory or a \*\* read only memory). The program instruction for performing the various functional components of this invention is accumulated in an internal memory 203 working. A processor 202 and an internal memory 203 can be used as the component according to individual, or single accumulation equipment like a use specification integrated-circuit (ASIC) chip. A processor 202 and an internal memory 203 consist of the electrical circuit for performing the various processing facilities described here.

[0077] The display 207 for displaying the cursor controller 206 for operating the external are recording equipment 205 for similarly storing the keyboard 204 for inputting an alphabetic character into a bus 201 and data and cursor and a visual output is connected. Although a keyboard 204 is a standard QWERTY keyboard typically, it can also be similarly made into a keypad like a telephone set. External are recording equipment 205 is immobilization, or can be used as the removable MAG or optical disk equipment. Typically, a cursor controller 206 has the carbon button or switch relevant to it which can program activation of a certain function. Furthermore, the audio output means 208 is connected with a bus 201. Typically, the audio output means 208 consists of a certain audio signal generating means (for example, digital-analog converter) and loudspeaker. The audio output means 208 can operate also as a player. Finally, the video outlet means 209 is connected with a bus 201. Typically, a video outlet means is the processing means of marketing for displaying a video signal on a display 207, for example, a video board.

[0078] Although the session access workstation is materialized on the system based on the computer which operates according to the programmed instruction which was accumulated in are recording data medium, this invention can be performed on the terminal of a fixed function, or the computer control mold display system of arbitration like LiveBoard.

[0079] A session access workstation operates with reference to time stream architecture using the program instruction for playback and control, as mentioned above. Drawing 3 is the simplified state diagram explaining fundamental low-level actuation of a session access workstation. When drawing 3 is referred to, a session access workstation is in the initiation condition 301 at first, when the request which reproduces a session is received. Subsequently, it goes into the session starting condition 302. The session which should be reproduced is chosen by the user between the session starting conditions 302. Temporary data of the selected session is searched and various players, a control unit, and an editor are synchronized. Various players should be noticed about the ability to also consider as the equipment which unified or divided them into the physical equipment same as a session access workstation. Subsequently, various windows (namely, a player, a playback control unit, and an editor) are displayed on the display connected with the session access workstation. It goes into the condition of having been named the "actuation" condition 303 at this time. The session access workstation is performing the interaction, and the operating state 303 described here is used in order to tell intention that the various functions of the user interface about a session access workstation can be called at any time. In any case, when it goes into the "actuation" condition 303 first, the queue for reproducing a session from the start is given to a session. While being in the "actuation" condition 303, various primitive functions are performed according to a user input. The call of a "playback" command or an equivalent graphic user interface input (for example, click on a specific icon) will cause initiation of a session (box 304). Thereby, a playback condition is set to "playback." The call of a "halt" command stops playback of a session and a playback condition is set to "a halt" (box 305). The call of the "display correction" command in a specific window or control makes a window correct (box 306). Such correction can change the truck displayed with a time-line interface. The call of playback control makes a reproducing point change (box 307). This gives a queue at a new point and causes playback of a session. The call of an editing command causes either correction to existing temporary data, or generation of new temporary data (box 308). The return to an operating state 303 arises by initiation of the actuation in various conditions.

[0080] Actuation of such a large number should be noticed about that it may be called, in order to attain a specific function on a session access workstation. For example, the actuation which chooses the object of a LiveBoard window can call correction" for "display following "a halt."

[0081] While being in operating state 303, the call of a "leaving" command makes an access session finally stopped

(idle state 309).

[0082] Drawing 4 explains the screen display of a session access workstation. The screen display of drawing 4 shows two or more windows currently opened to coincidence. About almost all the graphic user interface-oriented window, the size of a window and arrangement are a user's selection matters. It is displayed on coincidence the LiveBoard window 401, the video window 402, the time-line interface window 403, the note window 404-405, the window 406 of an audio, the multimedia edit window 407, and the meeting player window 408 that drawing 4 is referred to. The LiveBoard window 401 can operate as an editor as a player and a playback control unit. As a player, it is used in order to reproduce a LiveBoard time stream. As a control unit, it enables control to playback of a session through an interaction with the object displayed there (it states to details below). As an editor, a user can make the specific comment about the session which generates a new object or is displayed by the playback next to a session.

[0083] The video window 402 is a player used in order to reproduce the time stream of video. The note windows 404 and 405 are players used in order to reproduce the note which can be used as a laptop computer usable as capture equipment between the courses of a meeting, or the same equipment.

[0084] The meeting player windows 408 are a player and a playback control unit. using the event of temporary data of an and also [ it is versatility ], the meeting player window 408 offers the simulation of the dynamics of a meeting as it is reproduced. Playback is controlled by the interaction with the various icons which appear in the meeting player window 408. A meeting player window is stated to details below.

[0085] The audio window 406 is a control unit used in order to control the sound volume of the speaker for playback of the time stream of an audio, especially audio playback of a session.

[0086] The multimedia editor 407 is an editor used in order to generate the note which can be introduced and returned as a time stream at a session, an event, or other information. Furthermore, the multimedia editor 407 can be used in order to generate the multimedia document which sees and does not need the session access workstation of a sake.

[0087] The time-line interface window 403 is a playback control unit. The time-line interface window 403 is concretely expressed as they generate the view based on a temporary thing of an event between sessions. The time-line interface window 403 is stated to details below.

[0088] The time-line interface of an embodiment suitable at this time of this invention offers playback control by the interaction with the expression of an event in alignment with the time line. Any of a fixed period or a period indeterminate (between generate times is important) are sufficient as an event. Event data is loaded into a session access workstation during session starting. Event data is used in order to generate the visual indicator by which contraction scale adjustment was carried out on the time line. These visual indicators can be used as an index point of moving playback to the time amount to which a session corresponds directly. The internal structure for organizing event data to a time-line expression is described here.

[0089] The loaded event data is organized in order to make a user control event data by the easy method. This organization is defined by a mold map, an event menu, and the truck map. All event molds have the identifier of the layered structure called Htypes. for example, : -- the mold map which the Speaker/AdamLiveBoard/Edit/MoveButton/BettyNote/Laptop/Charlie user defined is used in order to generate such Htypes. A mold map is the set of the Ruhr which uses the parameter of each event and makes Htype. Subsequently, the set of Htypes is prepared into the event menu of the easy layered structure shown to a user in the label area of time amount truck area. All events can be displayed on a user on a truck. A layout is defined by the truck map. A truck map is a list of two or more trucks with the list of Htypes expressed on the truck with which each corresponds.

[0090] A visual indicator is generated about each Htypes. Although what kind of graphic style is sufficient as a visual indicator, most generally the class of two outlines of a visual indicator is used. The 1st class is called a segment. Since it corresponds to the period which an event produces, size adjustment is carried out on the time line, and a segment is displayed as a box of the rectangle by which contraction scale adjustment was carried out. The 2nd class is called a start point. A left end is located at right angles to the point of time amount that the event started, and since it means that the activity corresponding to an event in a right end continues indefinitely, these are displayed as a triangle "points out" out the right. About such an event, there is no interest about the period of an event (for example, event which shows that people begin to take notes). Various actuation can be performed about a visual indicator. Actuation of a visual indicator contains the following.

- Generate a visual indicator (a predetermined mold or mold by the default).
- Choose a visual indicator.
- Remove a visual indicator from selection.
- Move a visual indicator to a different truck (time amount is still the same).
- Copy a visual indicator to a different truck (time amount is still the same).

- See much more much information about the event expressed by the visual indicator.
- Edit the display parameter of the selected visual indicator.
- Edit visual indicator time amount.
- Adjust visual indicator start time.
- Adjust visual indicator (segment) end time.
- Adjust a visual indicator time amount location.

[0091] Drawing 5 explains the fundamental area in a time-line window. When drawing 5 is referred to, the time-line window 501 consists of two or more display area. The general-view time amount truck area 502 is used in order to offer a general view of a session, and there is less focal time amount truck area 503 than the whole session, or it offers much more detailed information over an equal period. It should be careful that actuation of each truck area can offer temporary information on the same mold. A difference is the scale of the details (namely, a part of whole session pair session) which information means.

[0092] The clock area 512 was defined in the general-view truck area 502, and the clock area 513 was defined in the focal time amount truck 503. In order to show the corresponding time amount span and corresponding scale of time amount truck area, each clock area 512 and 513 expresses clock time amount and a mark mark. The general-view time amount truck area 502 defined the special truck called the focal truck 508 in it. The focal segment 509 is on the focal truck 508. The focal segment 509 corresponds to the period of the session explained in full detail in the focal time amount truck area 503. Furthermore, label area relates to each truck area. Each label area is a thing which identifies the "truck" displayed in corresponding time amount truck area and which is accumulated and boiled. The label area 504 is equivalent to the general-view time amount truck area 502, and the label area 505 is equivalent to the focal time amount truck area 503. The relation between the general-view time amount truck area 502 and the focal time amount truck area 503 is shown by the focal area 507. It is made for this relation to look clear by the eye by the arms 510 and 511 which spread from the end of the focal segment 509 to the end of the focal time amount truck area 503.

[0093] Similarly, it is the carbon button area 506 which drawing 5 explained. The carbon button area 506 is the static area containing the "software" carbon button for calling various playback control functions. Such a carbon button is called by the method of pointing at and clicking by using a cursor controller. Arrangement and the function of a carbon button and it has the intention of offering easy access to the specific function generally used.

[0094] Used by it, since a carbon button 520 skips playback for 10 seconds and makes it return, a carbon button 521 skips playback for 10 seconds, and makes it postpone. The arrow in which the direction of the carbon button group 522 is shown is prepared for the migration according to the context, or scrolling. A carbon button 523 is prepared in order to suspend playback of a session. A carbon button 524 begins to reproduce a session. A carbon button 525 makes it possible to skip to the next event after the set of the selected truck or the selected truck. A carbon button 526 is prepared in order that close may correct to the edit mode a time-line interface or the visual indicator displayed on it. The carbon button group 527 is prepared for the deletion according to the context, or copy (for example, a visual indicator is copied or deleted). The carbon button group 528 is prepared the escape according to the context, or in order to fold up (for example, the set of the truck which folded up the set of a truck into the single truck, or was folded up is expanded). A carbon button 529 is used in order to display a session menu. A session menu enables him for a user to load data for a session, and to save a session, or to stop a session.

[0095] As described above, a part of session shown in focal time amount truck display area is shown by the focal segment contained on the focal truck of general-view time amount truck area. As for a focal segment, size is corrected by the user in order to change a focal span in focal time amount truck area. This is explained by drawing 6 -7. In drawing 6 , the focal segment 601 shows a part of session shown in the focal truck area 602. Other visual cues which identify this relation include the change to each clock area 603 and 604, i.e., the mark mark about time amount truck area and each initiation, size correction of end time, and a relative-position arrangement of time indicators 605 and 606. Similarly, in order to be shown in drawing 6 , it is the playback time indicator 607 in which the present playback time amount is shown.

[0096] Size correction of a focal segment is attained by various cursor control actuation. If the carbon button of a mouse is pushed in a suitable embodiment at present while cursor is located in the top at the right end of a focal segment, a focal segment will be extended towards migration of cursor. The left-hand side of a focal segment is also operated by the same method.

[0097] Since the span shown in focal time amount truck area is increased, drawing 7 is drawing of the result on the right-hand side of the focal segment 701 extended by the right. As for this span that increased, a twist is similarly reflected in change of the clock area 702 and 703 (change of contraction scale correction of a mark mark, and the end time about focal time amount truck area).

[0098] A time-line interface can be used in order to control playback of many sessions, or in order to double a focus with many spans of a single session. Drawing 8 is drawing of the time-line interface for two sessions. Two sessions may express one half in the morning one half of the meeting of all the day, and the afternoon. Or they may express the occurrence from which a meeting like a weekly project examination meeting planned periodically differed. Typically, two sessions have a certain similarity. When drawing 8 is referred to, the time-line interface window 801 consists of two general-view time amount truck area 802 and 803 and the corresponding focal time amount truck area 804 and 805. The focal segment 806 shows the details about the span in the session corresponding to the general-view time amount truck area 802, and its span in the focal time amount truck area 804. Similarly, the focal segment 807 shows the details about the span in the session corresponding to the general-view time amount truck area 803, and its span in the focal time amount truck area 805. An actual truck shows that it does not need to be the same in each focal time amount truck area 804 and 805. If they differ, separation label area is needed. Since the space of very many in display area is used, this is not desirable.

[0099] Drawing 9 is the further drawing showing the details of many spans of a single session being displayed on focal time amount truck area. This displays many focal segments on the focal truck of general-view time amount truck area. When drawing 9 is referred to, while the 1st focal segment 901 has the details shown in the area 903 where it corresponds in focal time amount truck area, the 2nd focal segment 902 has the area 904 where it corresponds in focal time amount truck area. Such a display is desirable when it turns out that a certain area without the interest to reproduce exists (for example, when it turns out beforehand that the argument on the specific page on LiveBoard was considered [ "hold" ]). Focal time amount truck area is equally divided into two focal segments on the same scale by the default. However, as drawing 10 explained, some focal time amount truck displays assigned at the session period are correctable. Here, it is increased to the area 1001 of drawing 10, and the area 904 of drawing 9 was decreased to the area 1002 of drawing 10 in the area 903 of drawing 9. The drag actuation which moves the left-hand side of area 904 to the left can perform this increment in size.

[0100] two user taking notes according [ drawing 11 - drawing 14 ] to following capture equipment, i.e., Liveboard, an audio recorder, and a personal computer, a videocassette recorder, and the carbon button for every user -- since -- change is shown on the time-line interface display about the system which changes. Of course, other combination of capture equipment may be used, and the event from some capture equipments is omitted and, thereby, serves as a different time-line interface display. The session participant about this example is named Adam, Betty, and Charley.

[0101] When drawing 11 is referred to, 1101 of general-view time amount truck area displays the information about the whole session, and has the focal segment which shows the whole session. Thus, the focal time amount truck area 1102 displays the details of the whole session. The playback indicators 1113 and 1114 show the current playback location in each of those period.

[0102] Furthermore, it is the session identifier 1103 that it is related to the session area 1101. A session identifier identifies the session reproduced.

[0103] The information connected with general-view session time amount truck area is offered, therefore the whole session includes the session start time 1104 and the session end time 1105. Each of the session start time 1104 and the session end time 1105 is relative time amount [ as opposed to / the actual time amount of the day when the session occurred can be shown, or / 0:0:0 ] ( drawing 11 explained like).

[0104] The information connected with the focal time amount truck area 1102 includes the span start time 1106 and the span end time 1107. Each of the span start time 1134 and the span end time 1135 is based on the actual time amount of the day of a specific span, or receives at the session start time 0.

[0105] The focal time amount truck area 1102 includes the truck label section 1111 for offering the information on what each truck expresses. Trucks 1117 and 1118 express the edit event performed on LiveBoard. A truck 1118 is generated according to the selection actuation performed in a LiveBoard window, and is stated to details below. A truck 1117 expresses the various edit actuation performed by the object on LiveBoard. In a truck 1117, each edit actuation has the visual indicator color-code-ized peculiar. Then, red expresses deletion of an object, green expresses addition of an object, and blue expresses migration of an object.

[0106] A truck 1119-1121 shows the event of an audio. A truck 1119 shows the segment about which "Adam" is speaking, a truck 1120 shows the segment about which "Betty" is speaking, and a truck 1121 shows the segment about which "Charley" is speaking. while the participant is talking, each visual indicator of a segment is visually distinguished by a color or the texture typical.

[0107] A truck 1122-1123 shows the start point of the note taken by Adam who is a participant at a meeting, and Charley. Notes were taken with the personal computer which is operating as capture equipment, or other text generation equipments.

[0108] A truck 1124 shows the instance of the event of "carbon button" press. Each meeting participant has the carbon button which can be pushed when a participant senses that it argues about some important things. Again, a visual indicator is color-code-ized in order to identify the person who pressed the carbon button. As for blue, Betty means having pressed her carbon button there, and green means that Charley pressed his carbon button.

[0109] A truck 1125 shows the instance of the event about a video time stream. Such an event can include taking action for people talking or carrying out a certain type about a feltboard of gesture. Such an event is identified by analysis of a video time stream.

[0110] As mentioned above, the selection truck 1118 is automatically generated as a result of the selection interaction in a LiveBoard window. The various objects (for example, language or drawing) as which a selection interaction is displayed on LiveBoard are just going to be chosen. Subsequently, the event about the selected object is displayed on a time-line window. Similarly, it should be careful that selection may arise about spatial area. In such selection, area is chosen and the event about the object arranged in the selected area is displayed on the time line. Since a part of session when marking was erased from LiveBoard is reproduced, especially this is useful (for example, the skip was performed between reproductive processes, and he has noticed that the object was erased, or it was replaced with another object.). If it returns and refers to drawing 11, various selection events will be explained. Again, suitably, a visual indicator is color-code-ized, in order to show the mold of an event. The generation event 1127 shows the point of time amount that the object was generated, the migration event 1128 shows the point of time amount that the object was moved on LiveBoard, as for the color change event 1129, it is shown that the object color was changed, and the deletion event 1130 shows that the object was deleted.

[0111] A user can correct the contents of each time amount truck area. A truck can be merged, added and concealed or can be moved. In order to correct the contents of time amount truck area, actuation is performed in a label section. For example, an audio track 1119-1121 may be chosen and merged by reference of drawing 11. It is explained by drawing 12 and the merged truck 1201 is generated by the result of such a merged truck there. Selection of a truck can be attained by operating a cursor controller, pressing a switch for the switch relevant to a cursor controller, and moving cursor so that the label of a truck may be passed. Directions of selection of a label are attained by a certain visual stimulus which expresses the label chosen by the video mode of reverse. Discernment of the selected label is attained by detaching a switch. Various truck actuation options are displayed at this time. Subsequently, a user chooses desired actuation (in this case, merge actuation). However, this is one method of performing truck display actuation. other technology -- a pull down menu -- or it can call with the command inputted from the command line.

- Actuation of Truck Generates - Truck Which Removes from Selection - Truck Which Chooses - Truck Containing the Following (Predetermined Htypes).

-- Which Rearranges Selected Truck - Which Folds Up Truck Which Deletes Selected Truck, and of which - Selection was Done (into One Truck) -- [0112] Which Edits - Truck Label into which Display Parameter of Truck Which Extends Selected Truck into Sequence of Truck the Htypes -- being Based, and of which - Selection was Done is Edited It produces that the point and click actuation on a different visual indicator differed from each other. The playback next to a time stream is set to the time amount corresponding to the beginning of a visual indicator in each case. In the case of a note (Notes), the window containing the text in a note can open. Generating of such actuation is explained by drawing 13. When drawing 13 was referred to, the visual indicator 1301 was pointed at and the switch on a cursor controller was clicked. This makes the text window 1302 containing the text generated between meetings opened (here, useful indication was made by text" Betty.). Those with the necessity of using this argument in a meeting with a superior official. "). The text window should be noticed about that it may appear similarly in the location where it differed outside time amount truck display area.

[0113] The time line should be noticed about the ability to use combining other visualization technology, in order to show the long string of the data which could display perpendicularly or horizontally or continued.

[0114] The form of a player is not limited to being reproduced with the completely same gestalt as the time of it of temporary data of a session being generated. The example of such a player is a meeting player. A meeting player is used in order to express various meeting dynamics so that it may be visible. This is useful to identify a speaker although the voice by which people are reproduced cannot be recognized. The meeting player of this invention simulates the activity generated between meetings. Drawing 14 explains the window 1400 which displays a meeting player. If drawing 14 is referred to, including the visual indicator 1401-1403 with which, as for a meeting, each expresses a meeting participant, visual indicators 1404 and 1405 will express note taking equipment (for example, laptop computer), and a visual indicator 1406-1408 will express a "carbon button." The icon which expresses the produced current event during playback is highlighted. Here, since the speaker expressed by the icon 1401 is talking, an icon 1401 is highlighted. Similarly, when someone pushes a carbon button, a carbon button icon is highlighted and displayed.

[0115] It should be cautious of it being possible to include a speaker's very small image in a speaker's icon. It is desirable similarly for the visual feature as the truck with which it corresponds on the time line that each associated icon is the same to be maintainable. Then, if the speaker icon 1401, the carbon button icon 1406, and the computer icon 1404 are connected with Adam who has the blue visual feature, these icons will serve as blue. However, it is admitted that there may also be a condition in which this is impossible for a limit of the color which can be displayed, or other factors.

[0116] Furthermore, the LiveBoard page, the title indicator 1410, and the clock 1411 are displayed on the meeting player window. A page and the title indicator 1410 show the "page" seen on LiveBoard now at this reproductive moment. A clock 1411 shows playback time amount. About the fact that a session expresses a meeting, a table 1409 is similarly displayed, in order to offer a visual cue.

[0117] A meeting player can also be used in order to change playback into a different speaker. This makes it possible to cross a meeting easily by the speaker of specification [ a user ]. The change to a different speaker is attained by the actuation "is pointed at and clicked" to a desired speaker's icon. This makes playback jumped to the following point in time about the selected speaker. Furthermore, this actuation acts to all icons and the jump to the next instance of the event related with the selected icon is performed there (for example, actuation "points and clicks" a page and the title indicator 1410 in order to skip playback to the beginning of the next page of LiveBoard is performed.). Thus, the meeting player of this example operates further, in order to control playback of a session.

[0118] A meeting player may be designed so that other various events which can be analyzed may be shown. For example, since it means when it laughed or applause has arisen, or in order to laugh or to skip to a next temporary occurrence of applause, an icon 1412 or the "laughter" "applause" icon 1413 can be offered. Furthermore, although a suitable embodiment is used in order to express a meeting, it is the same method and can express the activity of other molds, for example, a presentation.

[0119] The suitable LiveBoard window of an embodiment operates to coincidence as a LiveBoard time stream, a playback control unit, and a player of an editor at present. A LiveBoard window is based on the above-mentioned Tivoli software. A LiveBoard window operates by different playback mode. Namely, it is in the ANIMETEDDO mode which reconstructs the exact appearance of the past condition, and "BAUNSHINGU-ball" mode and "Naru (null)" mode in which cursor points at the area where marking/edit has broken out. In ANIMETEDDO mode, a LiveBoard window is in the recorded sequence and reproduces a LiveBoard time stream (namely, hysteresis list by which the time stamp was carried out). In "BAUNSHINGU-ball" mode, the condition of LiveBoard in the end (or exit status of a page) of a session is displayed, the event related to an object occurs, an object is highlighted, and cursor points. In "BAUNSHINGU-ball" mode, the form of cursor changes in order to show the mold of an event (for example, rubber for deleting the hand for moving the pen for generating an object, and an object, and an object). In the "Naru" mode, the exit status of LiveBoard in the end of a session (or exit status of a page) is displayed. NARUMODO is used when renewal of a LiveBoard window is not desirable (for example, it perplexes a user).

[0120] In each playback mode, the feature of the addition called a ghost leaves the faint version (the ghost) of an object on a display, also after it is deleted. This feature can be turned on and off by the user. A ghost object can be used as an index to the inside of a session.

[0121] Animation mode and BAUNSHINGU-ball mode show the mold of the event respectively produced in specific time amount. In BAUNSHINGU ball mode, although cursor tells the mold of an event, in full animation mode, animation shows the mold of an event by ANIMETO [ it ] how.

[0122] Playback control is produced by the interaction with the object displayed between playbacks. As an easy example, a user can choose an object and can call a playback command directly on LiveBoard (for example, time amount written on LiveBoard). The feature of other playback control is described below.

[0123] Since the function of all LiveBoard(s) can be used, a LiveBoard window can operate as an editor. Then, a user can operate these selves of the displayed object, or can add a drawing. Such actuation makes the time stream of these selves generated.

[0124] A LiveBoard window is important for operating as both a player and a playback control unit. With a known note taking system, a LiveBoard window offers the playback control means which is not strictly based on the exit status of note taking data medium on the contrary. A user can act on LiveBoard and mutual all the points between playbacks. He can notice that the user reproduced the session, and skipped to the point of them of a session (for example, the point and click actuation on a clock), and language was erased from LiveBoard there. This can carry out the trigger of the user, in order to consider a part of playback in relation to deletion of language in details.

[0125] Drawing 15 explains the LiveBoard window of a suitable embodiment at present. When drawing 15 is referred to, the LiveBoard window 1501 is performed on a session access workstation, and is the instance of the user interface

for Tivoli application. A LiveBoard window consists of the control area 1502 where both sides touch the object area 1503 in a boundary. Various control of the control area 1502 is active in the LiveBoard window 1501. An active and selectable object is marking displayed on the object area 1503. Similarly, the playback tool 1504 is explained. In the point which has relation in the selected object, the playback tool 1504 is used in order to begin playback.

[0126] The example of an interaction with LiveBoard for choosing an object on account of the generated hysteresis list, and asking the earliest point to it is possible. As mentioned above, a hysteresis list is a list of performed primitive actuation, the time stamp of each primitive actuation is carried out, and it is related with an object. The object was defined spatially typically and performed as a function. For example, a line is drawn from points X1 and Y1 to X2 and Y2. Then, when selection of an object is made, it is easy to see a spatial location, to find out the object of the location, to find out the primitive object relevant to an object, and to choose the time amount of the subsequently most remarkable event. The most remarkable event is a "generation" event when general. However, since other time amount may make a user cause interest, the selection actuation whose user can choose the most remarkable event (explained below) is offered.

[0127] A LiveBoard time stream can be generally classified into two molds, i.e., monotone and non-monotone. A monotone time stream mainly has a "generation" event. The happening only thing is what a new object is generated for (there are not migration, modification, and elimination). In a monotone time stream, a large majority of overwhelming objects have only one event relevant to it (the generation event).

[0128] A non-monotone time stream consists of generation, modification, migration, and a deletion event. Thus, each object can have one or more events relevant to it. It is useful to distinguish change (namely, migration) from which the location of an object changes, and change which is not so. If the time stream does not have any location change events, each object has the only location in a display top. If there is a migration event, an object may occupy much locations during a session. Finally, if a time stream has a deletion event, an object may disappear by a certain time amount between sessions.

[0129] A Liveboard window is a stand-alone or can be used as a playback control unit correlated with the time-line interface. The correlation with a time-line interface is described below. In stand-alone mode, selection of an object and the call of a playback tool (playtool) will cause the playback in the time amount of the "generation" event about an object about a monotone time stream (this is because it is the only event connected with the object).

[0130] It is what "event" in the case of a non-monotone time stream, a problem uses, in order to start the playback time amount of the selected object. If a user points at an object with a playback tool, a system will reproduce the session of the stage of the reasonable early event relevant to an object. However, other time amount about an object may be important. For example, the time amount to which the object was moved may be important. There is an option which accesses other event time amount about an object. For example, the event menu which made all the events relevant to an object the table is offered. The playback whose user only chooses a desired event starts in the time amount relevant to an event.

[0131] the technology of an exception is using the event [ degree ] playback carbon button which skips playback time amount to the following event relevant to a current object. Still more nearly another technology using time-line functionality is expressed below.

[0132] The idea of the clock in a Liveboard window on LiveBoard was mentioned above. Since a clock offers temporary reference of when the activity broke out in the Liveboard window, it is useful in the case of playback of a session. The clock should remember that it is the graphic object generated by clock gesture. A clock can be operated and edited in the same way as all graphic-arts objects. However, the clock object is special and it includes a special property in the data structure expressing its "internal" time amount. [ of itself ] If a playback tool touches a clock, a reproducing point will be set to the internal time amount of a clock, and will not be set to the time amount (for example, between the generate time) of some events relevant to a clock. Therefore, a clock can be used in order to deal with it as time amount which became independent of the hysteresis of itself. By the default, the internal time amount of a clock is the same time amount as generation actuation of a clock. However, it comes out so, a certain necessity is not, and this draws various useful functions.

[0133] The internal time amount of a clock is changeable. At present, if edit gesture (it points now and holds) is performed on a clock in a suitable embodiment, the menu into which a user can change the internal time amount of a clock will pop up. These change can be made into relative (for example, only N second moves internal time amount to the front or back), or either [ being absolute (that is, internal time amount being moved to the specific time amount T) ]. Since the former is much more exact when adjusting slightly the internal time of a clock, in order to attach an index, it is the most useful.

[0134] For example, in a meeting, it takes up and argues about an interesting problem. The user of LiveBoard can attach

an index to this discussion by generating a clock and writing a note. However, after discussion starts, a clock is usually generated to some extent behind. Therefore, a clock can be edited, in order to attach an index to initiation of discussion much more correctly and to adjust the time amount to hard flow slightly. This edit can be carried out between access sessions between meetings or in the back. Since the latter is convenient to reproduce a meeting from current clock setting in order to distinguish whether setting is exact, it is the most useful. In current operation, facilities are given to the user by making playback start, after edit is performed, and feeding back the validity of time amount setting to a user to a user immediately.

[0135] A user can generate a clock between access sessions by adding a clock via a LiveBoard window (seeing the LiveBoard window as an editor hereafter). However, it sets in this condition, and the internal time amount of a clock is set to the current reproducing point of a playback tool, and is not between that generate time in an access session (it is not useful at all). For example, it thinks of the user who accesses a still earlier meeting. He hears the audio of a meeting and hears the important idea which was not able to attach an index between meetings. Then, in respect of [ this ] playback, he generates a clock and, subsequently to that next door, types a note. This clock is a new index to a meeting. Therefore, the quality and the amount of index addition can be strengthened in the access time by enabling it to generate the object holding the time amount in an original meeting.

[0136] It is often desirable to see all the events relevant to an object. For example, a user may desire to get to know the existence and the stage of modification of an object, or deletion. As mentioned above, all the events relevant to the object chosen by LiveBoard are displayed on the selection truck of a time-line interface. This offers the means for displaying the index to the inside of a session with the relation which the user specified by the functionality between LiveBoard and a time-line interface.

[0137] Other features of the interrelation by the interaction with a LiveBoard window are explained by the scenario of the following described about drawing 16 - drawing 20. The expression of a LiveBoard window and a time-line interface should be noticed about being simplified. Especially the control area on a LiveBoard window and the general-view time amount truck area of a time-line interface are not shown in drawing.

[0138] Reference of drawing 16 wants conversation when this language is written for a user to look at language in the LiveBoard window 1601, and to be reproduced. A user chooses the "redundancy" of language 1602 using a suitable selecting technic (gesture which surrounds language with a circle here using a pen/mouse). Subsequently, the time-line interface 1603 is updated on a selection truck, in order to show all the events connected with the selected language. Here, a visual indicator 1604 is displayed on the selection truck 1605. A visual indicator 1604 expresses the event which writes in language. the event of others [ this example ] -- language -- [0139] which is not associated as it is "redundant" Similarly, the playback time indicator 1606 is explained by the time-line interface 1603. Then, playback should be shortly noticed about it being the point after [ "after" ] language "redundancy" was written.

[0140] Drawing 17 explains the synchronization between a LiveBoard window and a time-line interface. A user moves the playback time indicator 1703 within the time-line interface 1702 just before a visual indicator 1604. Subsequently, the LiveBoard window 1701 is updated in order to display exactly what suited on LiveBoard then. When especially compared with drawing 16 , language "redundancy" should be noticed about not appearing in a LiveBoard window.

[0141] If a playback command is called, both a LiveBoard window and a time-line interface will be updated synchronously. After a while, language "redundancy" appears in LiveBoard.

[0142] Here, a user notices that some events were on the time line just before the event on a selection truck. A user chooses the visual indicator expressing these events on the time line, and the element with which the drawing corresponds is immediately highlighted in a LiveBoard window. That is, the interrelation of a drawing and the time line is bidirection.

[0143] This is explained by drawing 18 . A visual indicator 1804 is chosen from the time-line interface 1803. In the LiveBoard window 1801, language "router (router)" 1802 are highlighted (here, the circle of a dotted line showed). Therefore, the event corresponding to a visual indicator 1804 is related with language "a router."

[0144] Next, a user looks at a thick box on drawing of a Liveboard window. And a box is chosen. Three visual indicators are displayed on the selection truck of a time-line interface (or three visual indicators can be highlighted on an edit truck). The visual indicator of the 1st mold shows the generation event for a box, the visual indicator of the 2nd mold shows the Rhine width-of-face modification event, when a box is made thick, and the visual indicator of the 3rd mold shows the event of migration of the box on a drawing. As described above, the visual indicator after a time-line interface is visually distinguished, in order to show the class of event which they express. A user is interested in the reason nil why a box is thick, moves a time amount marker to the Rhine width-of-face modification event, and calls playback using a playback tool.

[0145] This is explained by drawing 19 . A box 1902 is highlighted in the LiveBoard window 1901. A visual indicator

1904-1906 is highlighted with the time-line interface 1903. As mentioned above, each of these visual indicators expresses the specific mold of the event relevant to an object. Furthermore, in a suitable embodiment, each visual indicator has a color at present on a certain visual property of corresponding to the mold of an event, and a type target. [0146] A user starts either playbacks or chooses a visual indicator 1904.

[0147] Next, a user considers drawing on another page of a drawing. Between meetings, this drawing is drawn repeatedly, is erased, and is redrawn, and a user wants to hear why a former version was not considered to be the right. A user chooses the area of a drawing in which drawing is located. Here, some sections of the time line are highlighted and the event about all the objects that suited the area of drawing containing the object deleted or moved becomes clear. A user moves a time amount marker to the visual indicator of deletion with which the 1st was highlighted, and finds the argument of why the 1st version was changed.

[0148] The visual cue which the object came to become in a specific location is a ghost object. A ghost object is the faint outline of the object used in order to show that the object existed in this specific location in the point of a certain time amount during a session. the user has noticed various object ghosts and what was there -- or it wants to get to know whether it will be (for example, list of items with which it becomes clear by the ghost object that the item was erased in the end of a session).

[0149] This is explained by drawing 20. When drawing 20 was referred to, the user chose the area 2002 of drawing of the LiveBoard window 2001. If drawing 16 is referred to simply, area 2001 is equivalent to the list of language, "addition", a "router", and area including "redundancy." The time-line interface 2003 has the visual indicator 2004-2006 of a large number highlighted here. These are related to "the event corresponding to the event relevant to area 2002, i.e., language, "addition", and a router", and the event related redundantly. This should be cautious of it being only spatial correlation, i.e., the event expressing the language erased from the list having been highlighted similarly. Furthermore, the "ghost" object 2007 is explained to drawing 20. 2007 was erased for the object of the location shown by the "ghost" object 2007 in respect of [ this ] playback.

[0150] As mentioned above, an editor allows a user to add a time stream and event data after the fact. It is marking on LiveBoard (or added new marking) when a LiveBoard window is used as an editor or what is edited. In this case, addition to a LiveBoard time stream and/or deletion are performed as a result of such edit. Moreover, another time stream is also generable.

[0151] When used as an editor, a LiveBoard window offers a very effective means by which the session when the non-participant was recorded can be examined efficiently. Partially, this is because it can use during playback of all the fundamental capacity of LiveBoard of a session. This is best described about drawing 21 and drawing 22 which show the LiveBoard window in the end of the session of both before and behind an editing session. LiveBoard of this example is operating by the BAUNSHINGU ball playback mode.

[0152] If drawing 21 is referred to, marking of various handwriting has appeared in the LiveBoard window 2101. As mentioned above, marking of these handwriting of each can be used in order to attach an index into a session. However, between a certain meetings, some of marking may be unrelated (for example, mark 2102), or it may be hard to decipher them (for example, mark 2103). As [ accomplish / to LiveBoard / furthermore, / although there is an important point about which it argued / nothing ]

[0153] Drawing 22 explains the LiveBoard window of drawing 21 after someone edits. When drawing 22 was referred to, the edited LiveBoard window 2201 was extended in order to make it adapted for the notes accomplished with the editor. The edited LiveBoard window 2201 inserts a text using a clock, and offers much more much an index and information. In drawing 22, since it meant what is discussed by the point, the text was related with each clock. Since it means what a mark 2103 means, a text 2203 can be used there. A clock / text 2202 means what collection of marking expresses, and when generation of marking started. Clearly, the edited LiveBoard window enables the next viewer to use a LiveBoard window much more effectively as a playback control unit.

[0154] As an editor, marking which exists on a LiveBoard window is changeable. This can be performed in order to erase the unrelated marking 2102, for example, marking of drawing 21, so that the next viewer's of a window attention may not be distracted.

[0155] Another aspect of affairs of time stream architecture is control to the gained time stream. the time stream from which the user was gained -- especially -- an audio and video -- like -- " -- it is very interested in the time stream which is record of "literally. They are interested in who accessed their activity and literal record of conversation. They are interested in they being recorded when and getting to know \*\*, and they desire control to this. After the argument which was not recorded on the other hand although it was interested, a user will fly, if it carries out and they are recording it, and he is \*\*\*\*\*.

[0156] User control is supported by time stream architecture. However, operations are between the captures of a session,

and after that [ both ], and need the user interface tool for giving a user control effectively.

[0157] the handling with a "control event" special [ time stream architecture ] -- providing . A control event is - time amount span (time amount and a period are made to start) which specifies the following.

- Class of Time Stream-Control Which Should be Controlled (Purge Level of Protection)

- It is [0158] when Control Becomes Effective. Supposing the gained session can have many time streams, a control event can control them alternatively. For example, it is useful to control only the audio between the delicate spans of an argument. A control event can also specify all the time streams of a session.

[0159] There is some kinds of control. The recorded data can be physically purged from a data base. Access to whether the recorded data are heard and it seeing can be refused, or can be given to the group of a specific user or a user. Access can be used as a different mold like access which can only be read, access which can append new data, access which can edit data completely, or access which can change an access control. For example, only the chairperson of a meeting can edit meeting record, only a group's manager can change an access control and a control event can append it to the data by which only the participant at a meeting had new data gained, and as the audio from which only other individuals of a specific group were gained has been heard, it can be specified.

[0160] Finally, the timing of control can be specified. This is the most useful in order to purge. if the acquired record describes a detail -- immediately (for example, in order to erase an argument of the whole personnel) -- immediately after a session (for example, in order to erase the meeting which does not excite interest) -- or it can purge to different time amount like [ of a number of days or how many months after ] (to for example, document arrangement and storage management sake). However, to a non-participant, access to meeting record can be refused for one week, and it can also be specified that it gives the time amount of editing record into the chairperson.

[0161] Although time stream architecture enables a control event to specify various control, architecture does not carry them out directly but represents the control to the various components of architecture. In order to carry them out, it can code appointed [ many of ] in a data base. A time stream player and a time-line interface also serve as assistance which carries out a certain access control similarly. Purging must usually be carried out according to a recorder or a certain process which understands the data format of a specific time stream. The process to purge can be made into either of whether record is compressed whether overwrite of the data which should be purged is carried out, and by keeping outside the data which should write in again and should actually be purged taken out "it burying by zero." Although time stream architecture does not guarantee perfect operation of a control event in itself, it performs the best thing which can be performed with an available component in the given condition.

[0162] It turns out that it is generally very delicate to offer an access control by the method based on time amount. For example, since delicate decision is dealt with, I consider purging the audio / video record of the argument on the span for 10 certain minutes, and think that video record shows the feltboard of a wall. Even if the span for 10 minutes is purged from video, there will be no guarantee that the remainder of the discussion does not remain on a feltboard after the span. However, the control based on time amount is useful as a practical problem at almost all time amount.

[0163] The control event is useful, only when a user can generate them easily and can edit them. For a user, it is natural to specify a control event with a time-line interface after a session.

[0164] At present, in a suitable embodiment, in order to give a user control of the duplicate eye to record of the audio during a session, the easy set of a user interface tool is offered. It is important to make a user notice when record was started. A red light on a video camera performs this. About Liveboard, while record is performed, "under record" is displayed. [ blinking ] A user can turn a recorder on and off for both with the pause carbon button and record carbon button which are a hard carbon button on physical equipment, and the software carbon button after LiveBoard or a computer display. However, this control must be expected. That is, in order to consider as any of record and a pause, it must not be in a user whether it is interested or delicate, but he has to know beforehand whether it is \*\*. In fact, this is not effective. That is, a user cannot often recognize these conditions after they start until it becomes behind enough. Therefore, posthook (post-hoc) control is needed.

[0165] Our user interface offers this by carrying out the concept of "record ON" and "record OFF" (adding to a pause and record), and this enables a user to control the record after the fact by the method suitable during a session. This idea is that the whole session is actually recorded, a user can mark whether it is record OFF or it is record ON on the portion of a session, and the portion of record OFF of record is purged after a session.

[0166] If it is made to start in order that a session may record, the blinking "record ON" cautions will be displayed together with a record off-carbon button. If a record off-carbon button is pushed, a user is this time or can mark that record is off on a session at the time of N quota. For example, if an argument turns to delicate subject and a debate person writes in a note after several minutes, they can set in [ of the last used as record OFF ] 5 minutes. Similarly, when the session is not recorded, blinking "record OFF" is displayed together with a record on-carbon button, and,

thereby, a user can mark that record is ON on a session. For example, when discussion becomes interesting, the last 10 minutes can be returned after record to origin. These control insists to the end of the session when a user can determine whether the whole session should be recorded.

[0167] These user interface tools' carrying out is making a control event generate between sessions (a record off segment is marked noting that it is purged, and a record on-segment is marked noting that it permits access.). After a session closes, a recorder is called, in order that the process which analyzes the control event made about consistent assignment (that is, the span of a session is marked on both record ON and OFF) may be performed and a process may subsequently purge record. One technology for confrontation solution is granting priority further to the latest control event. However, a user looks at a control event via a time-line interface, and before he calls a purge, he can ask that they are edited.

[0168] It is that it is difficult for a user to know whether one difficult point of this user interface will return in order to change record. One amelioration to this will use the audio tool for analysis of us who give the time amount point a user has a meaning further. For example, after pushing a record off-carbon button, the menu of change of the backing up point or speaker to the pause of the latest audio can be expressed. The user was reproducible from these, in order to determine a right time amount point.

---

[Translation done.]

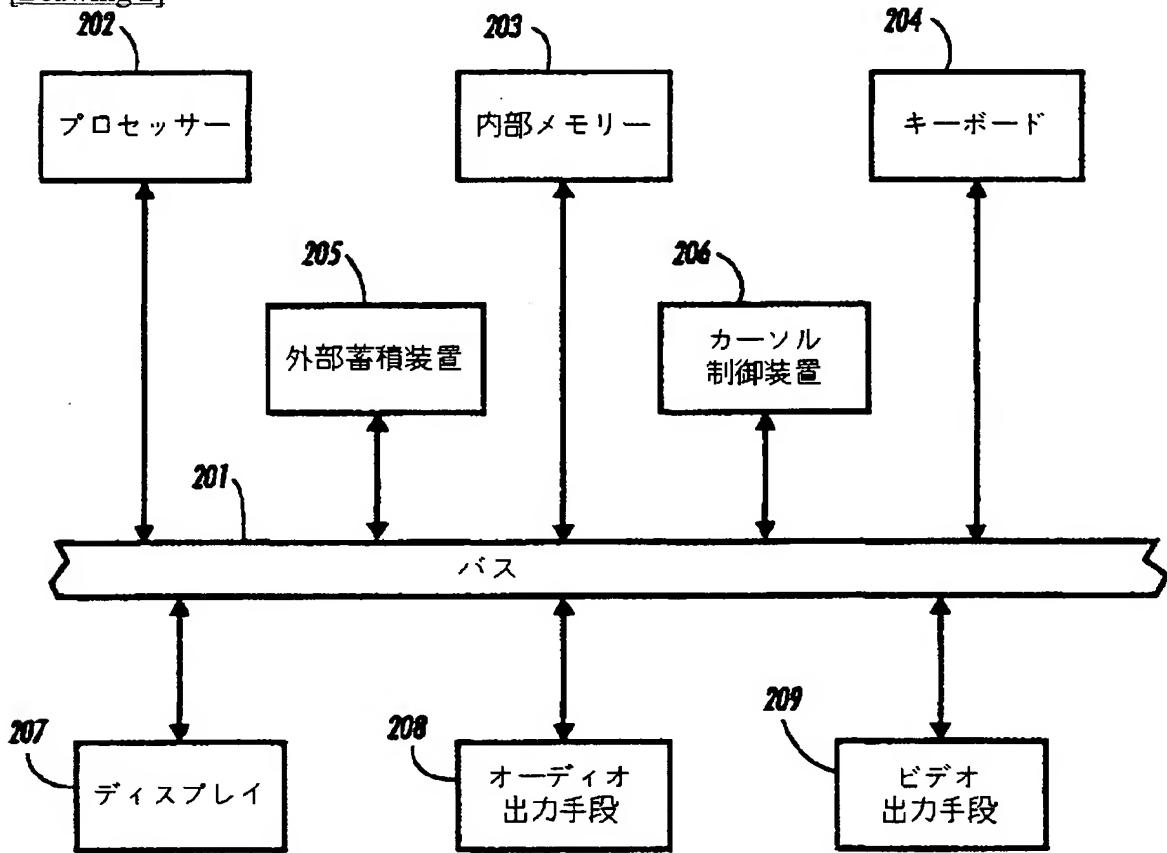
## \* NOTICES \*

Japan Patent Office is not responsible for any damages caused by the use of this translation.

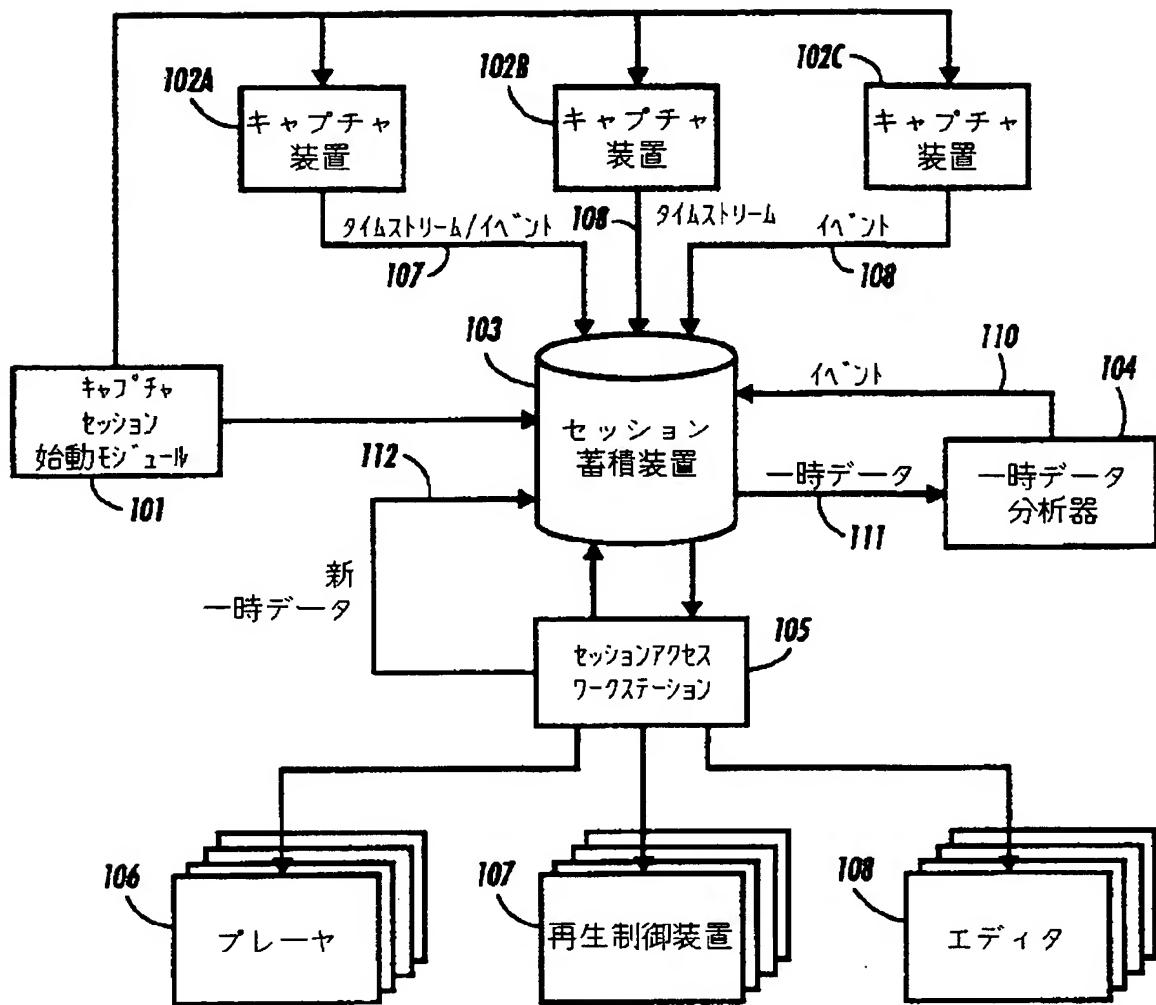
1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

## DRAWINGS

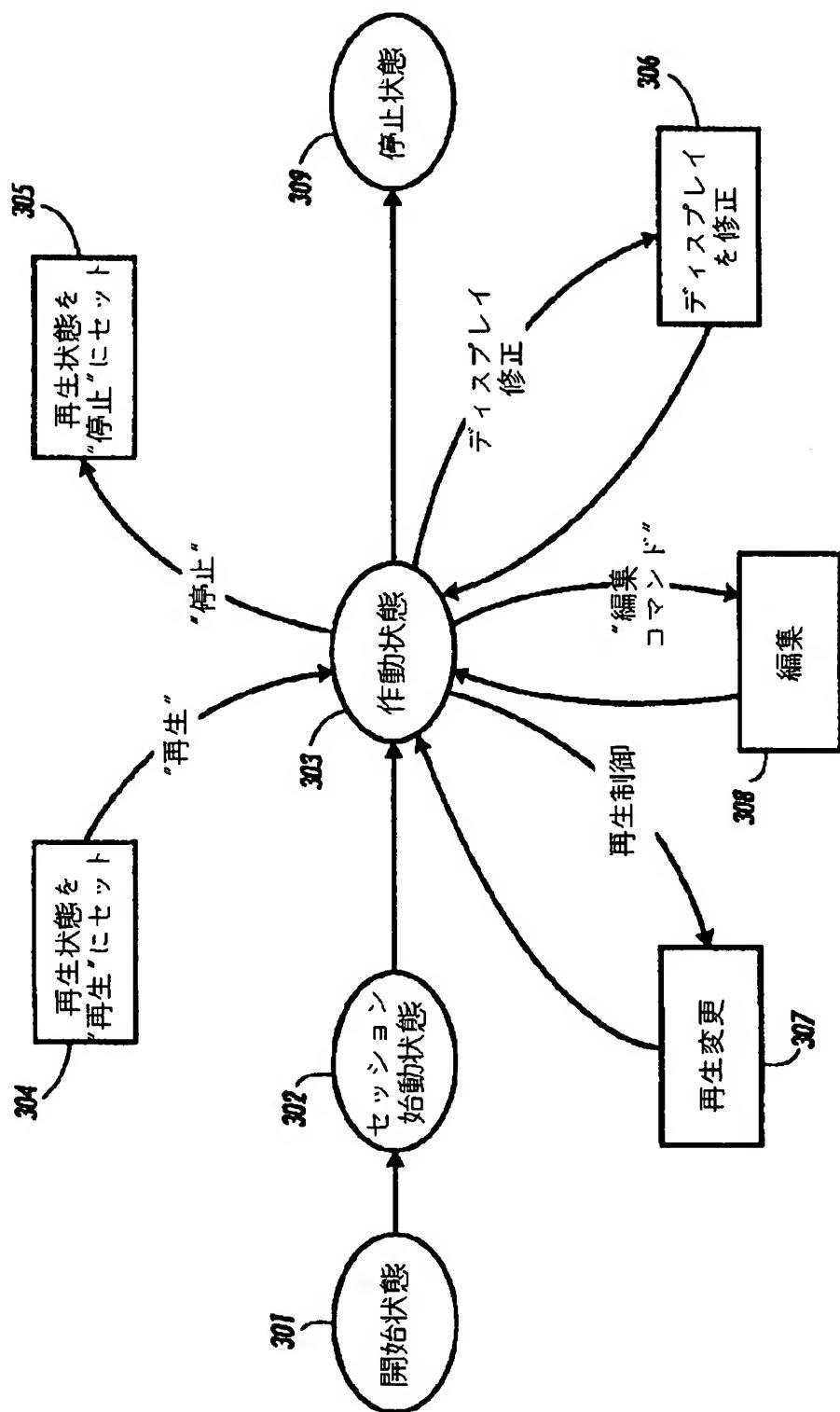
## [Drawing 2]



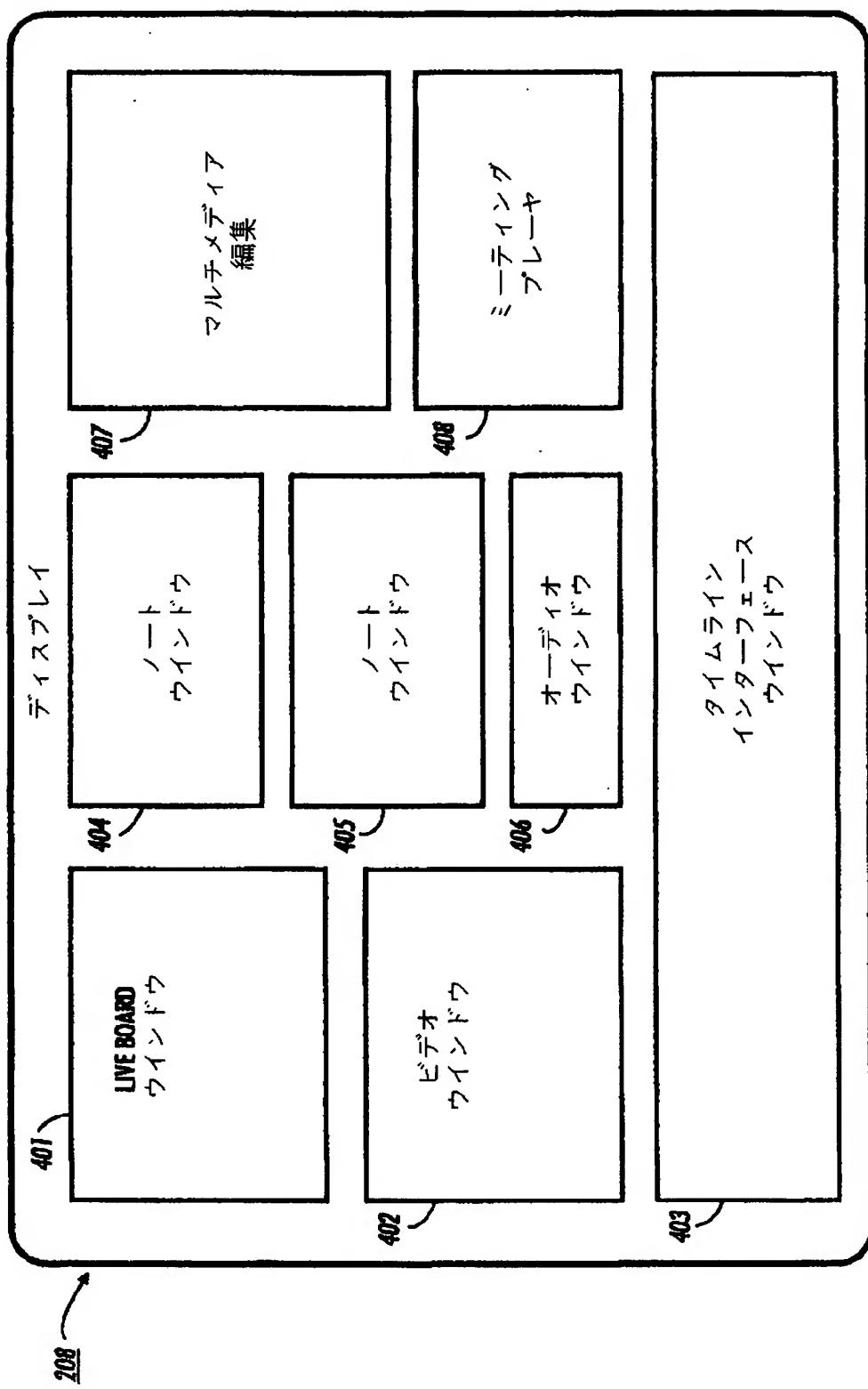
## [Drawing 1]



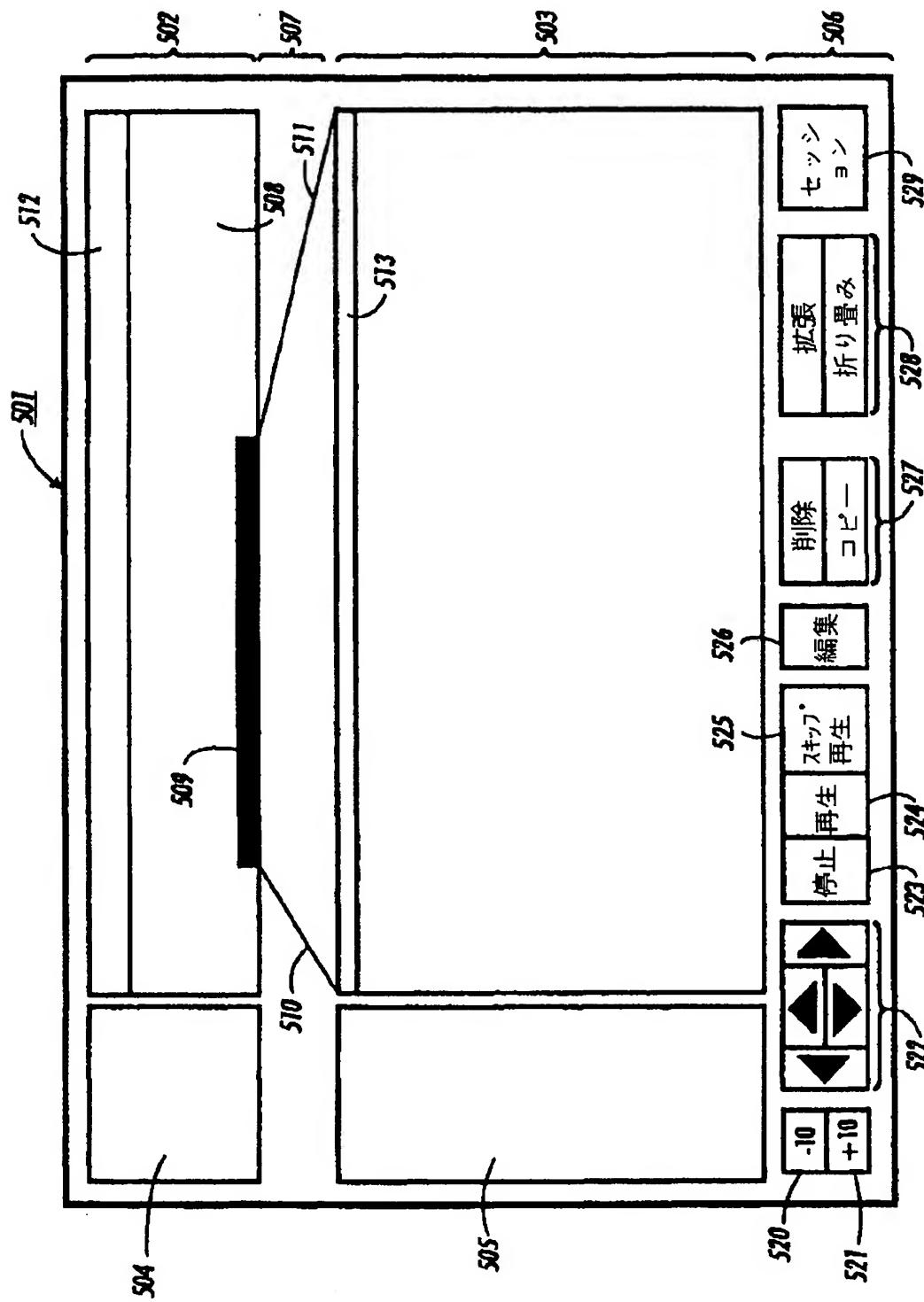
[Drawing 3]



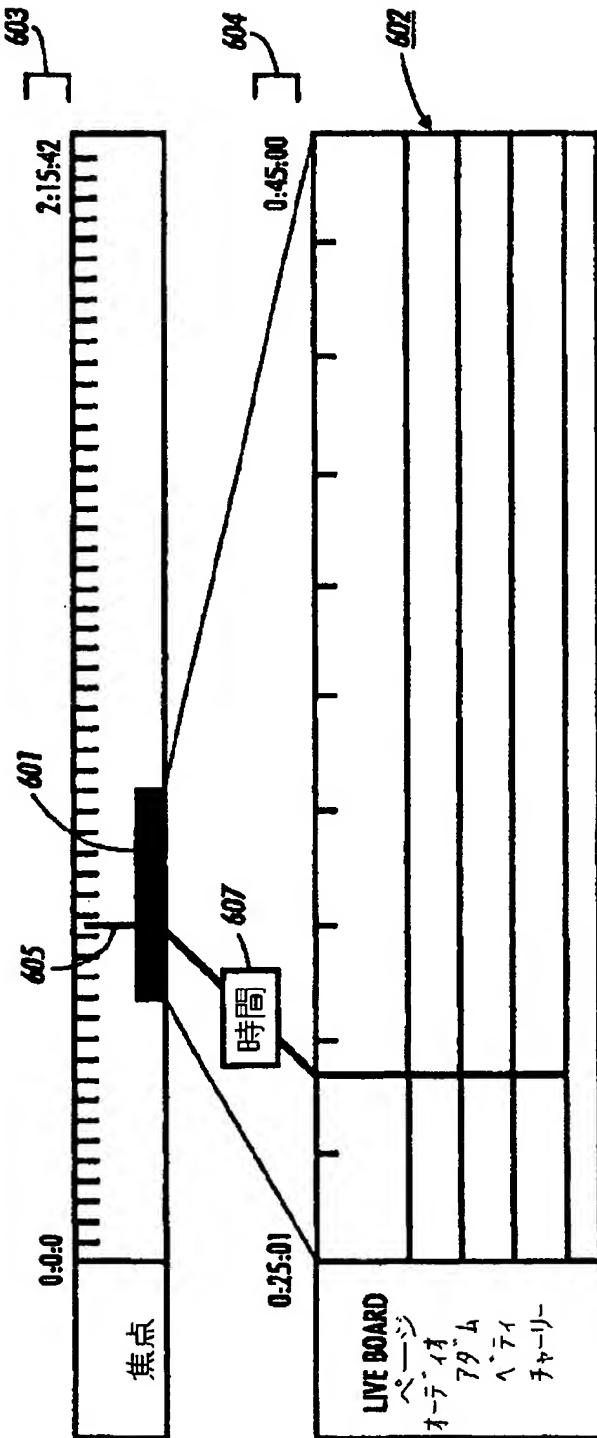
[Drawing 4]



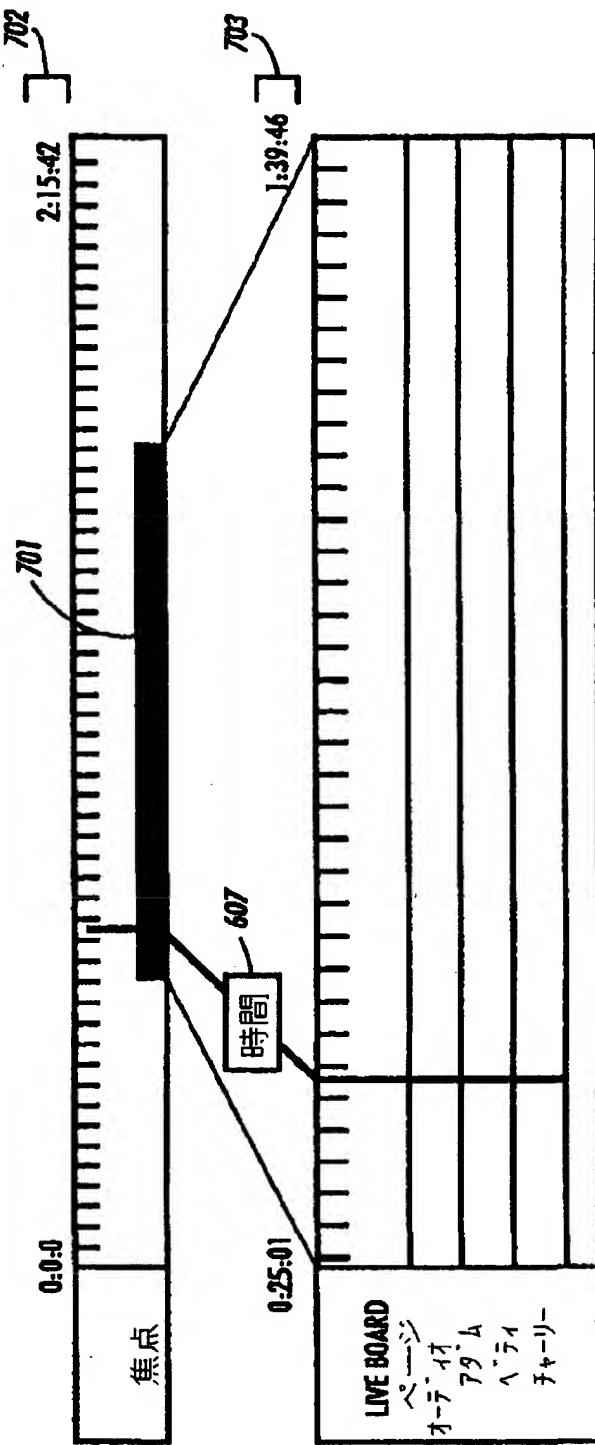
[Drawing 5]



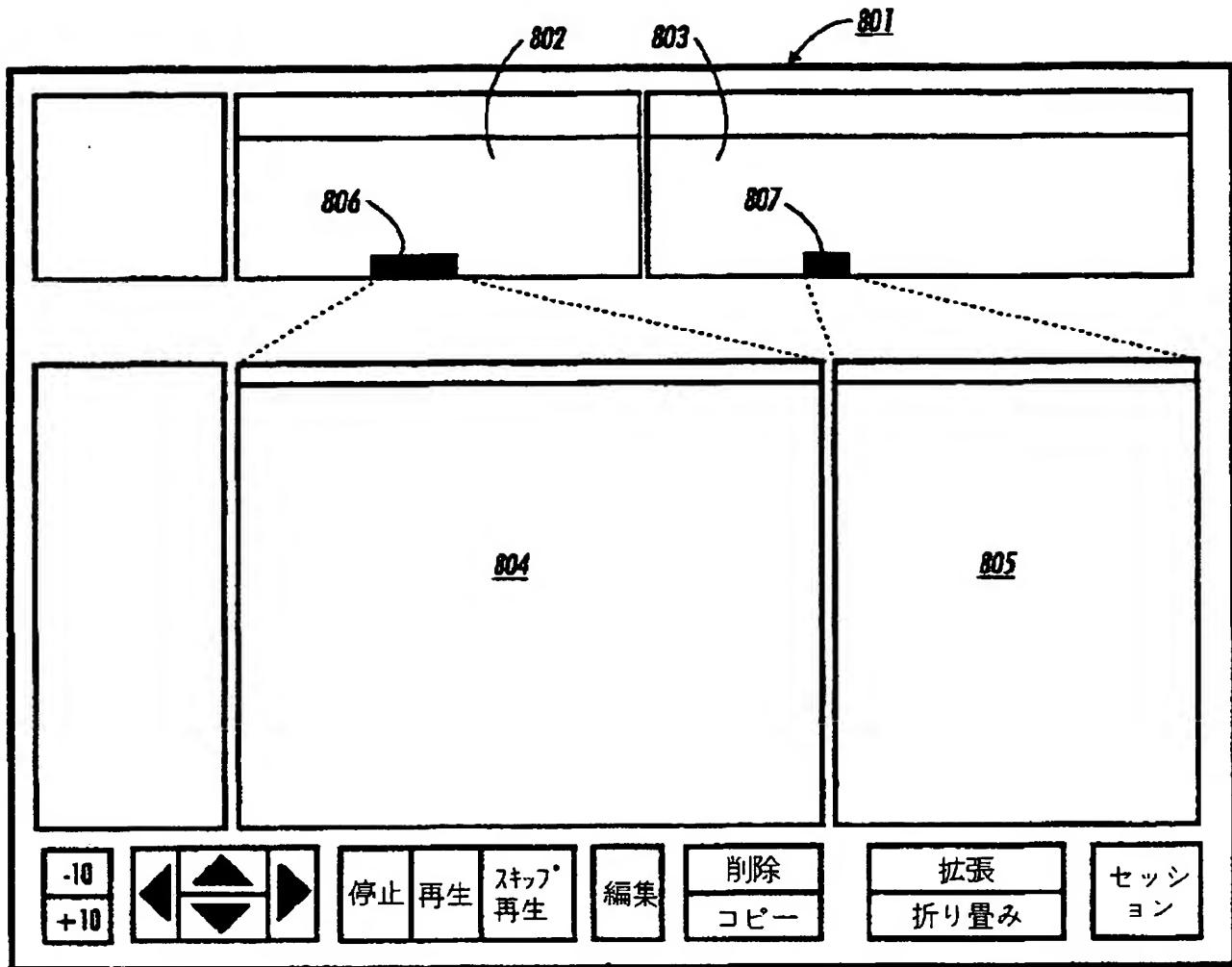
[Drawing 6]



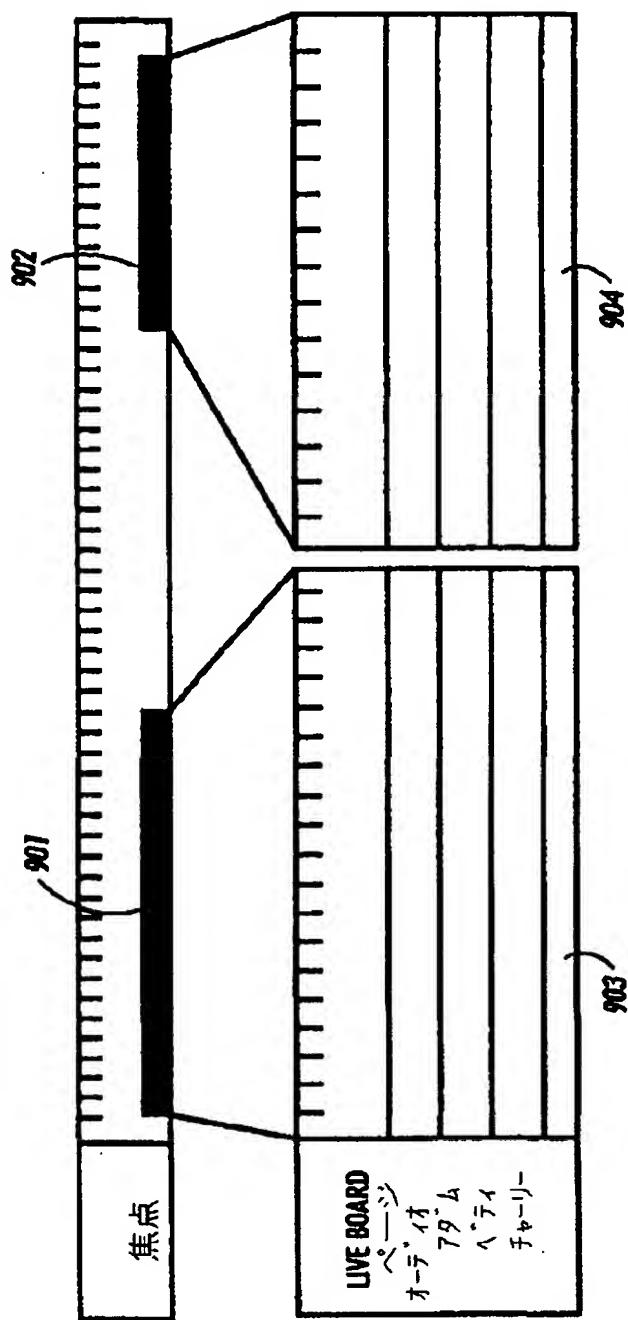
[Drawing 7]



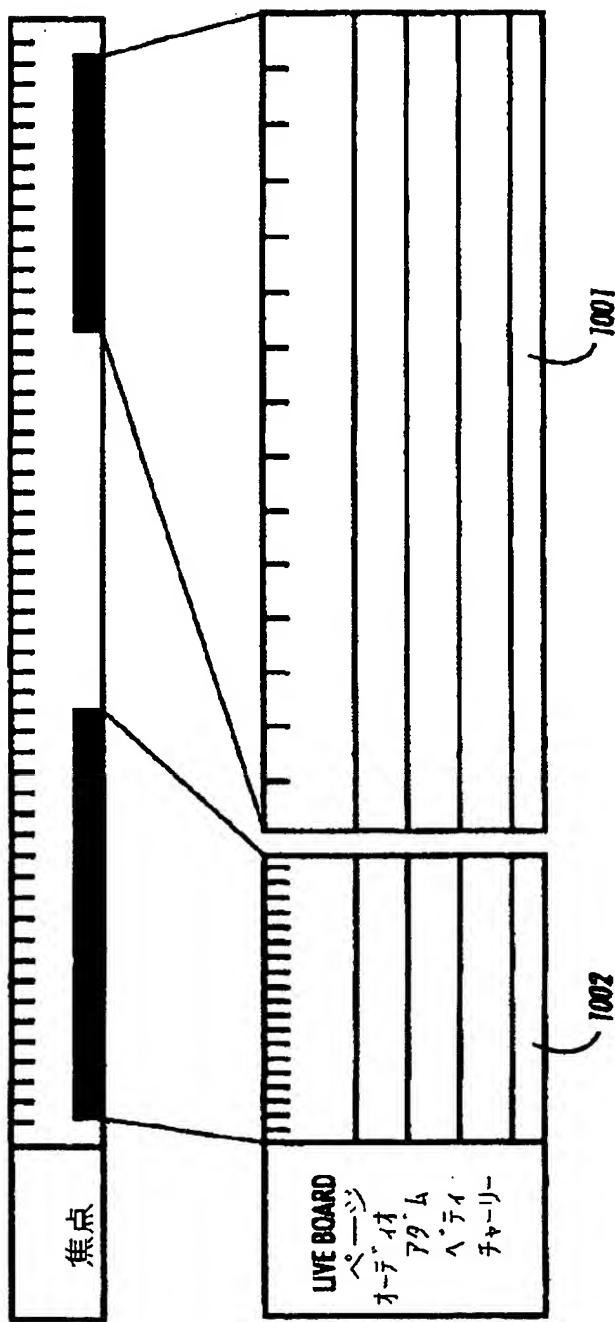
[Drawing 8]



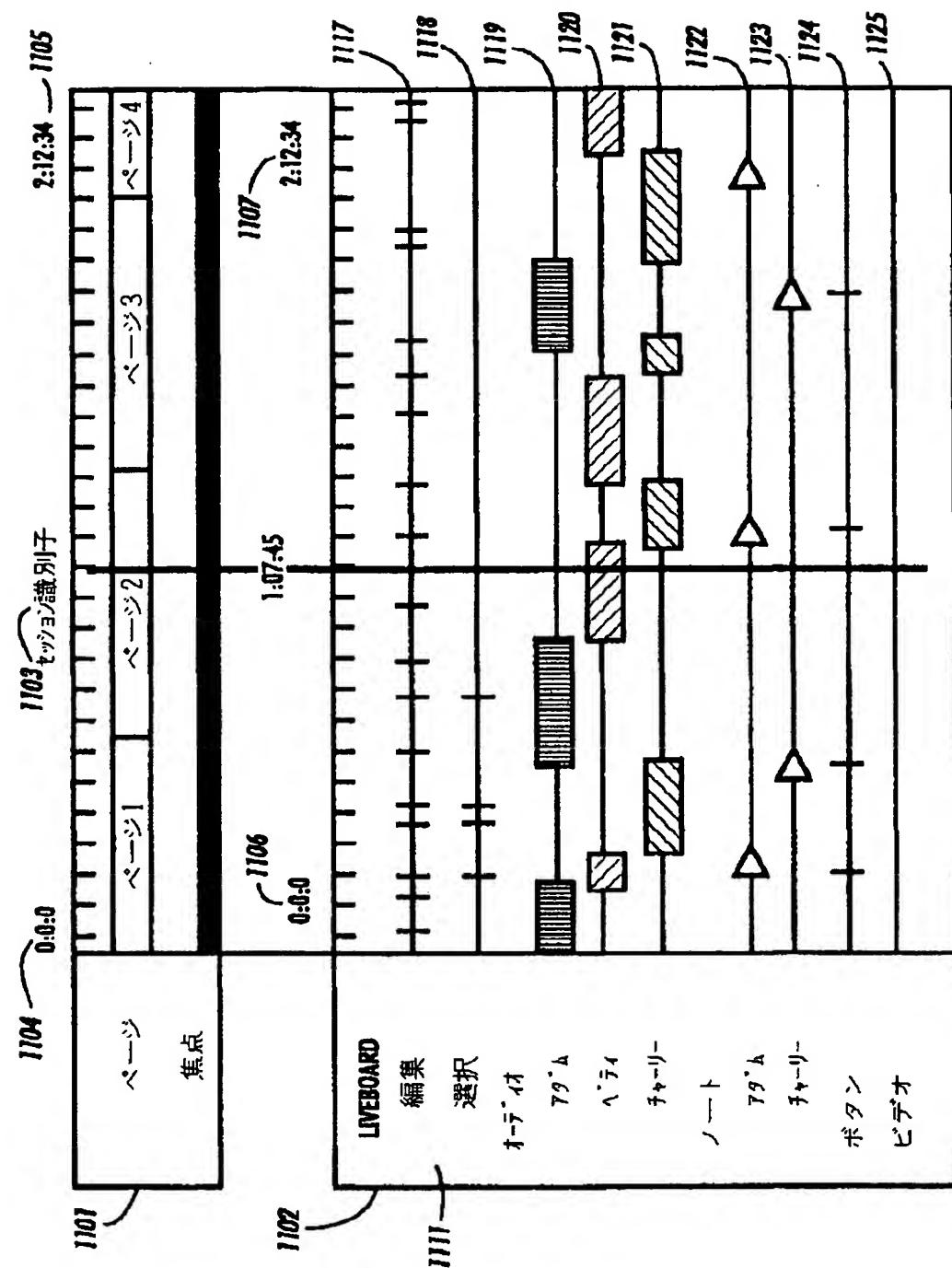
[Drawing 9]



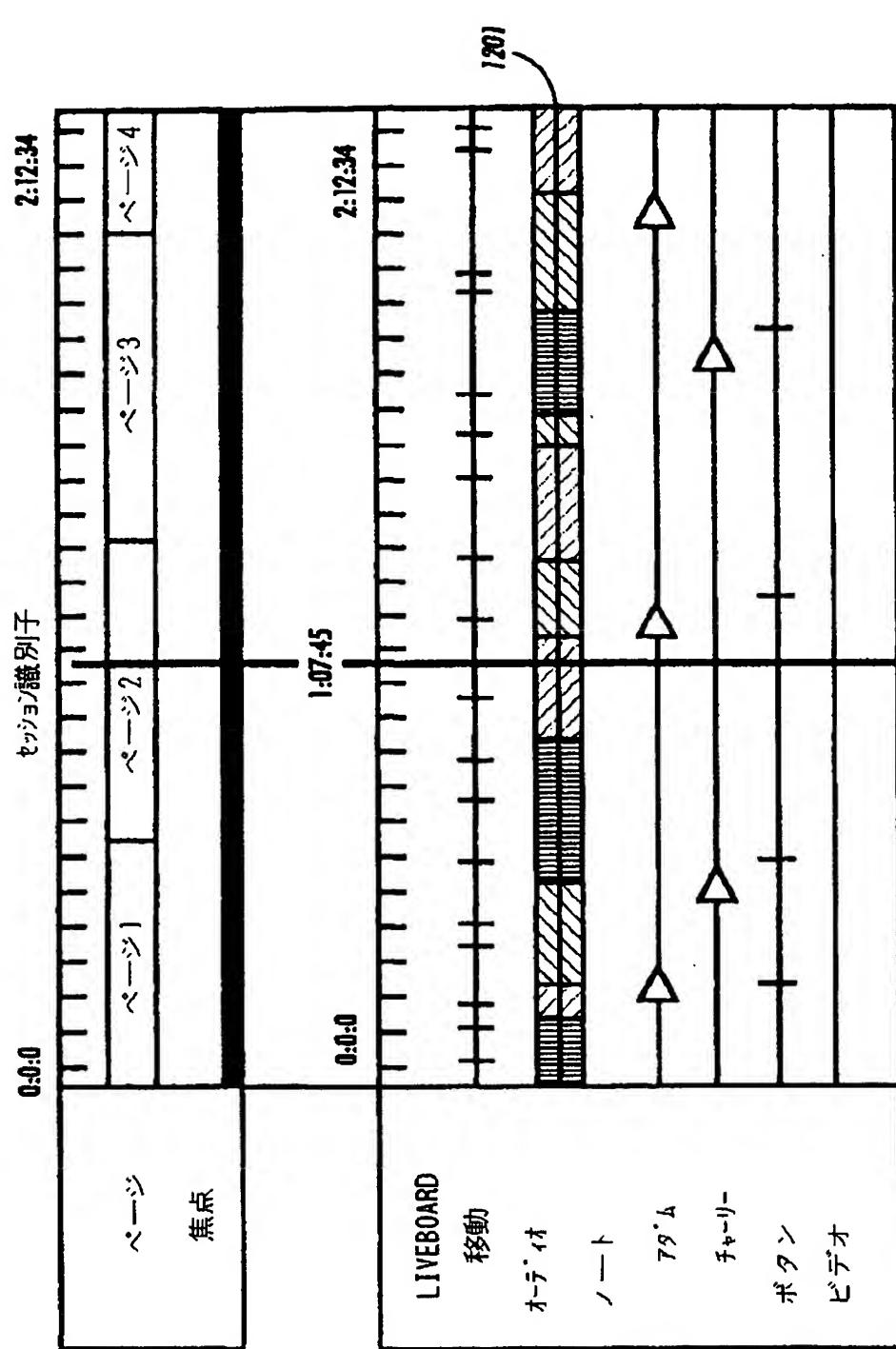
[Drawing 10]



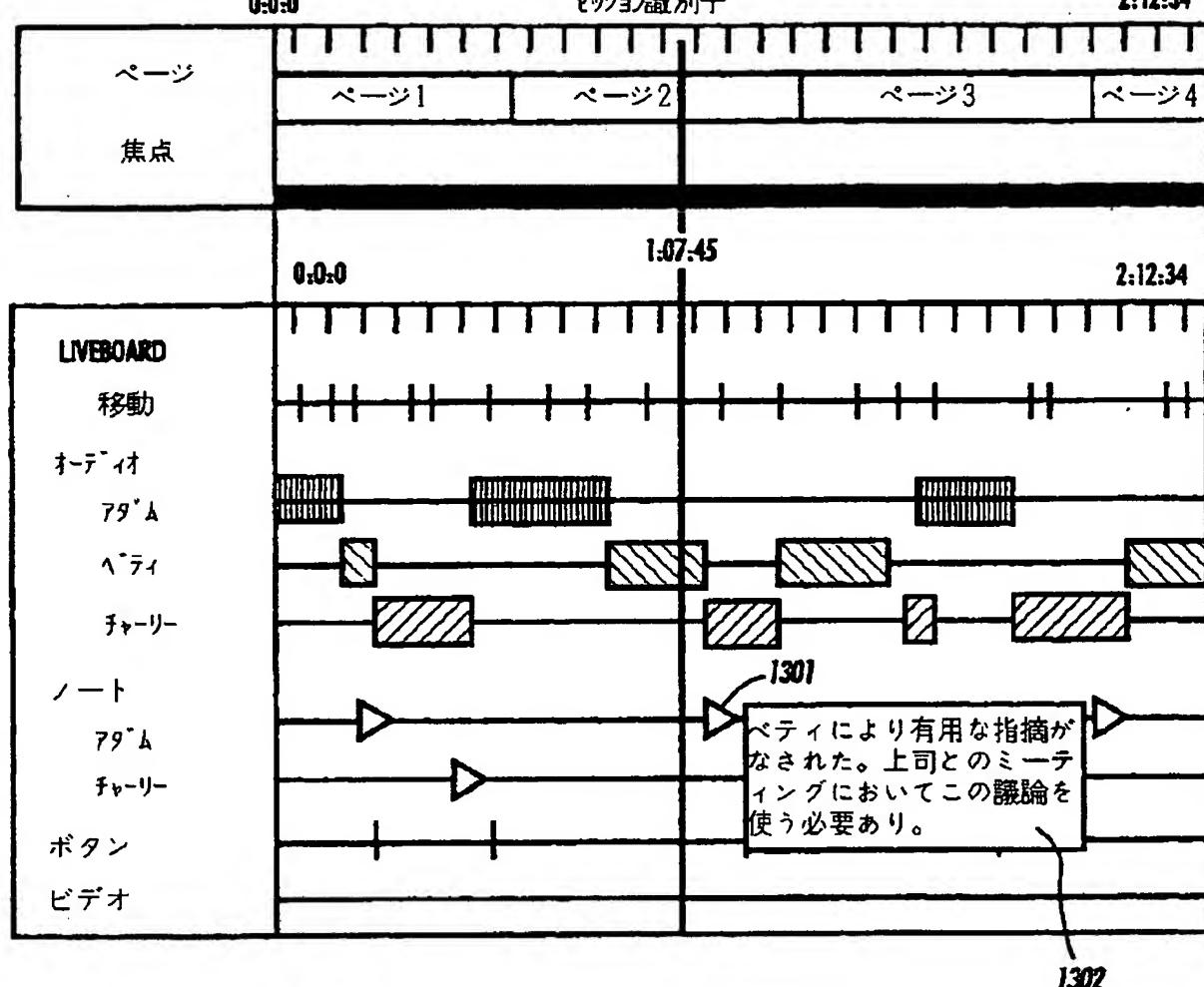
[Drawing 11]



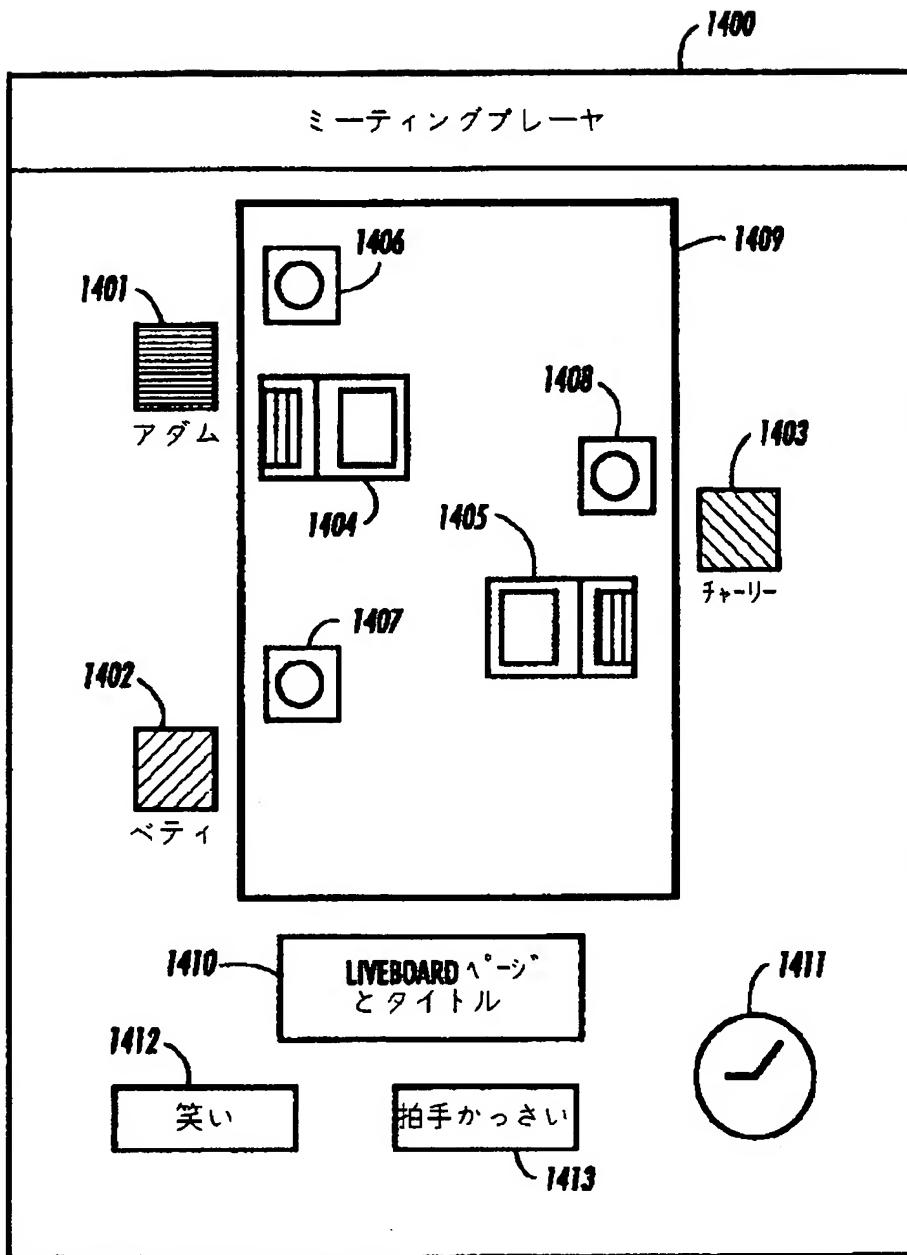
### [Drawing 12]



[Drawing 13]

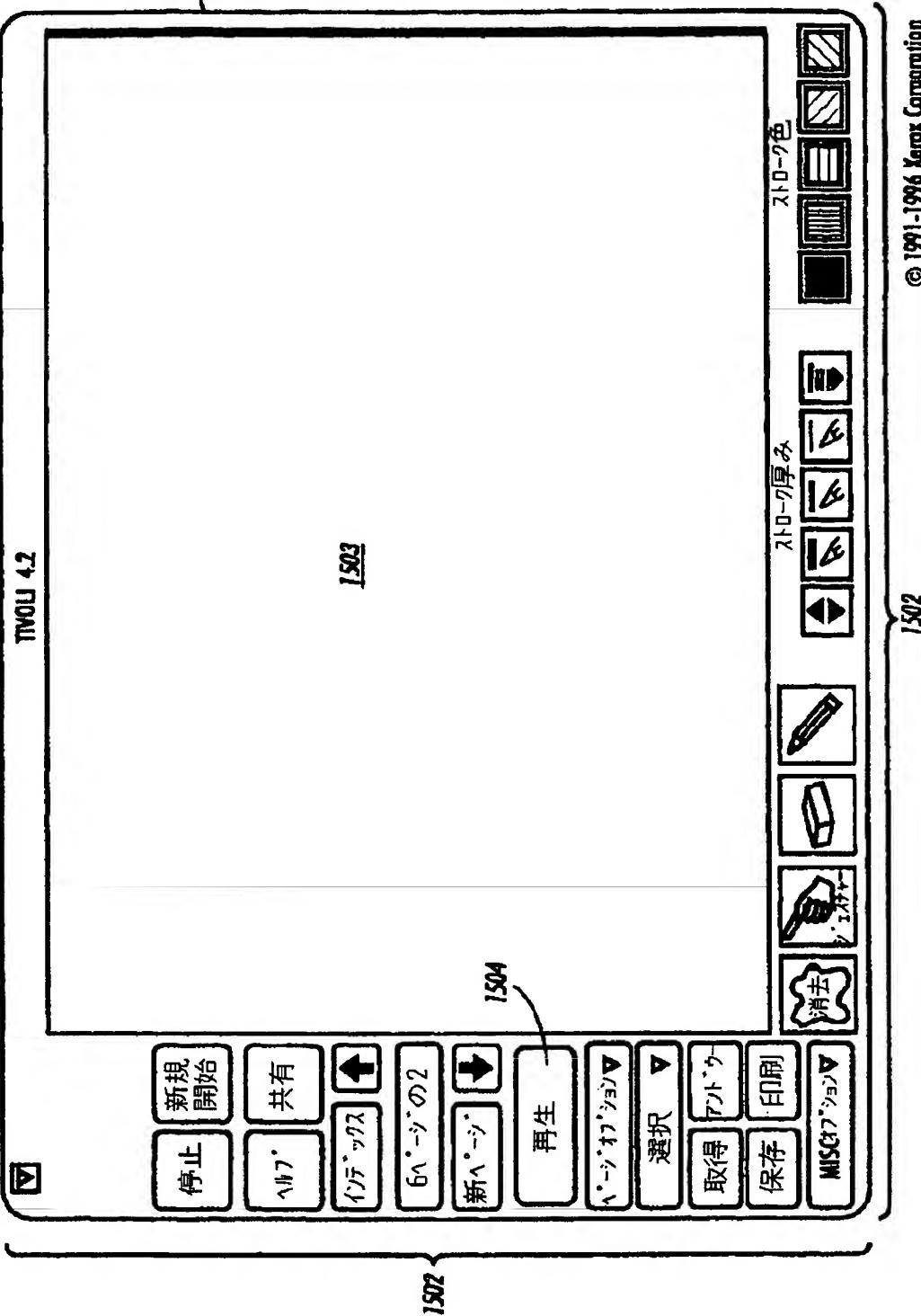


[Drawing 14]



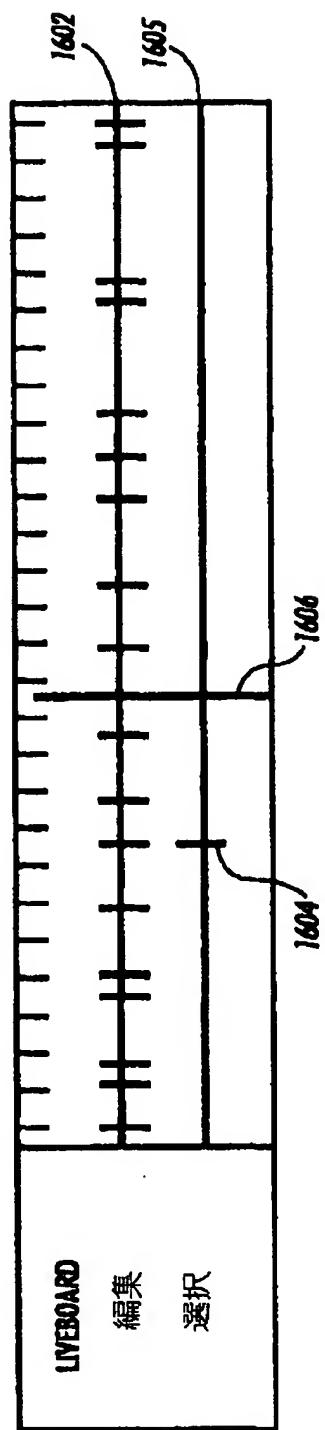
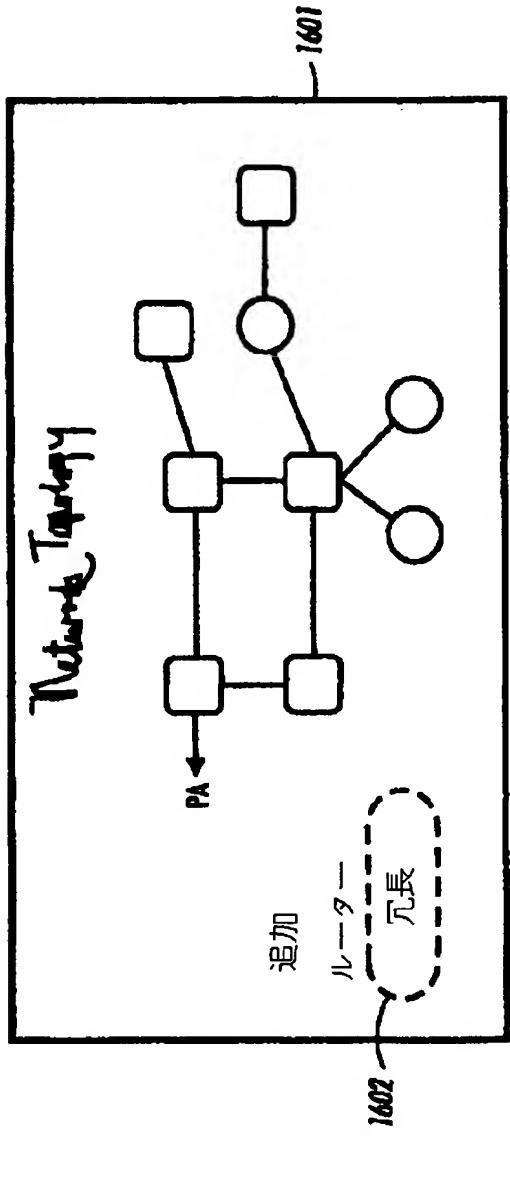
[Drawing 15]

1501

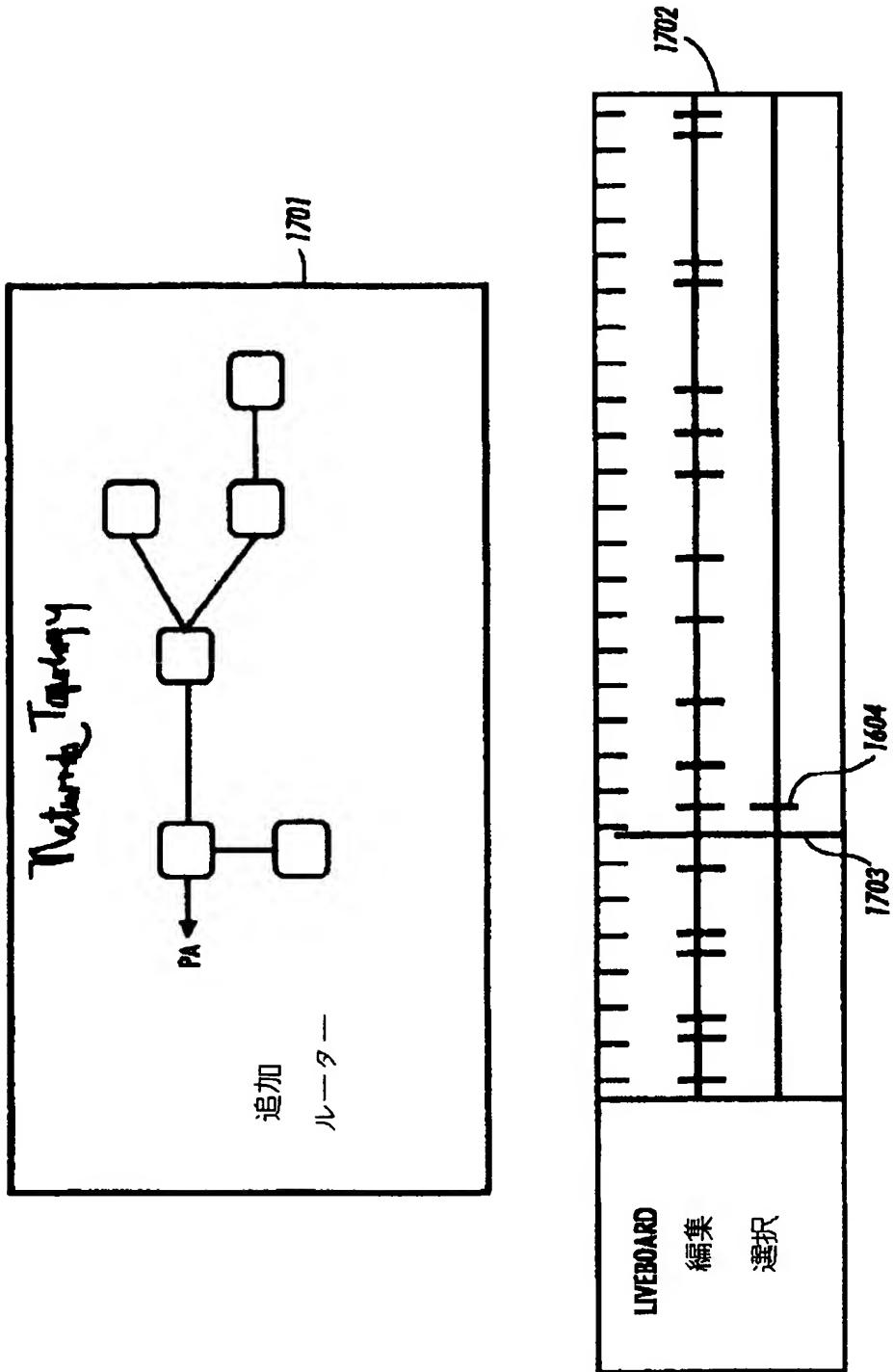


© 1991-1996 Xerox Corporation

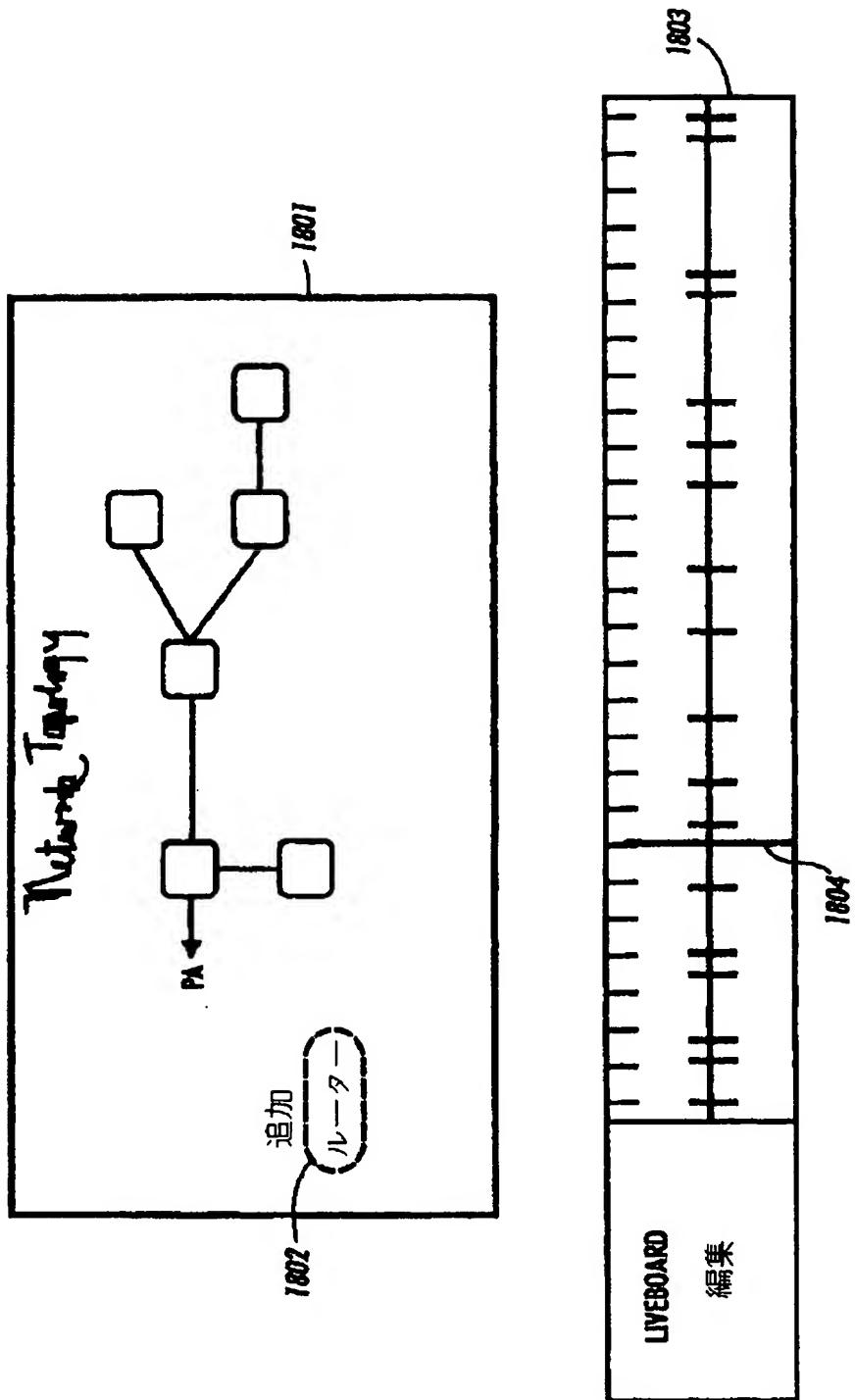
[Drawing 16]



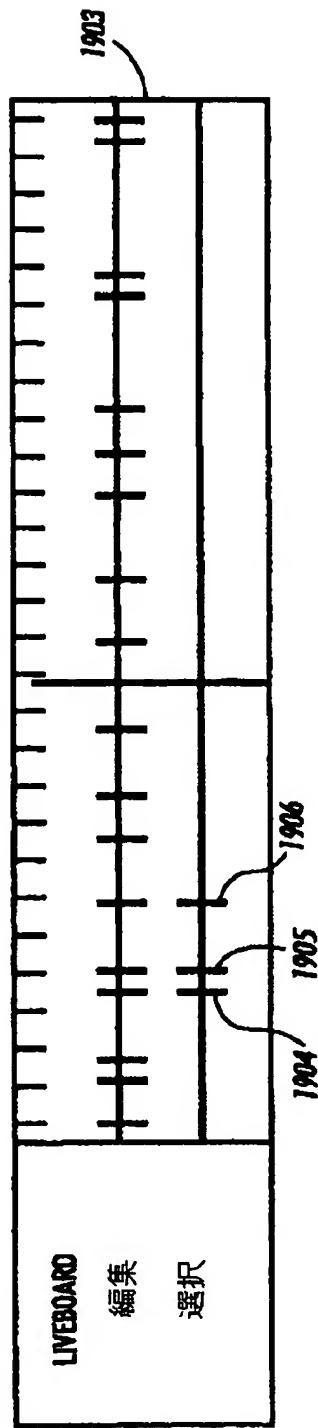
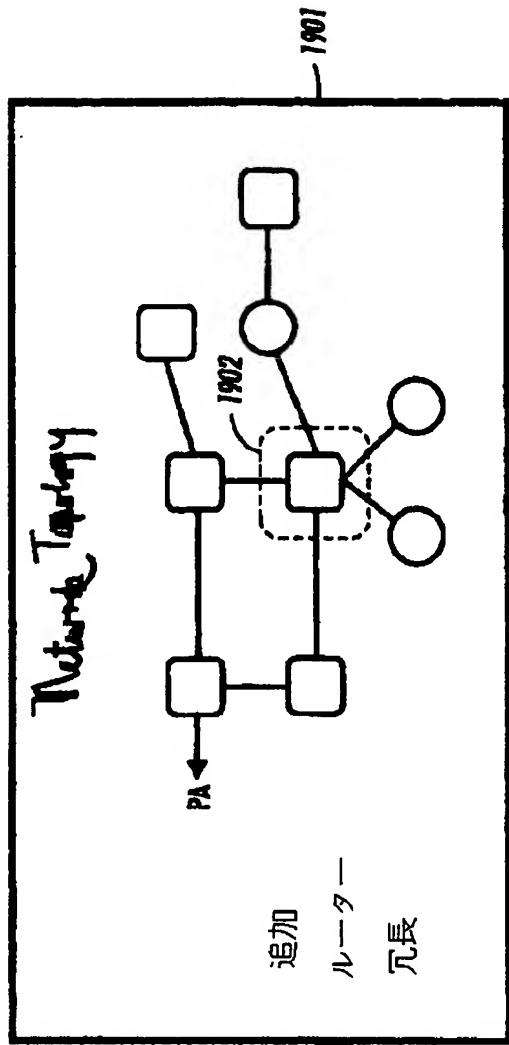
[Drawing 17]



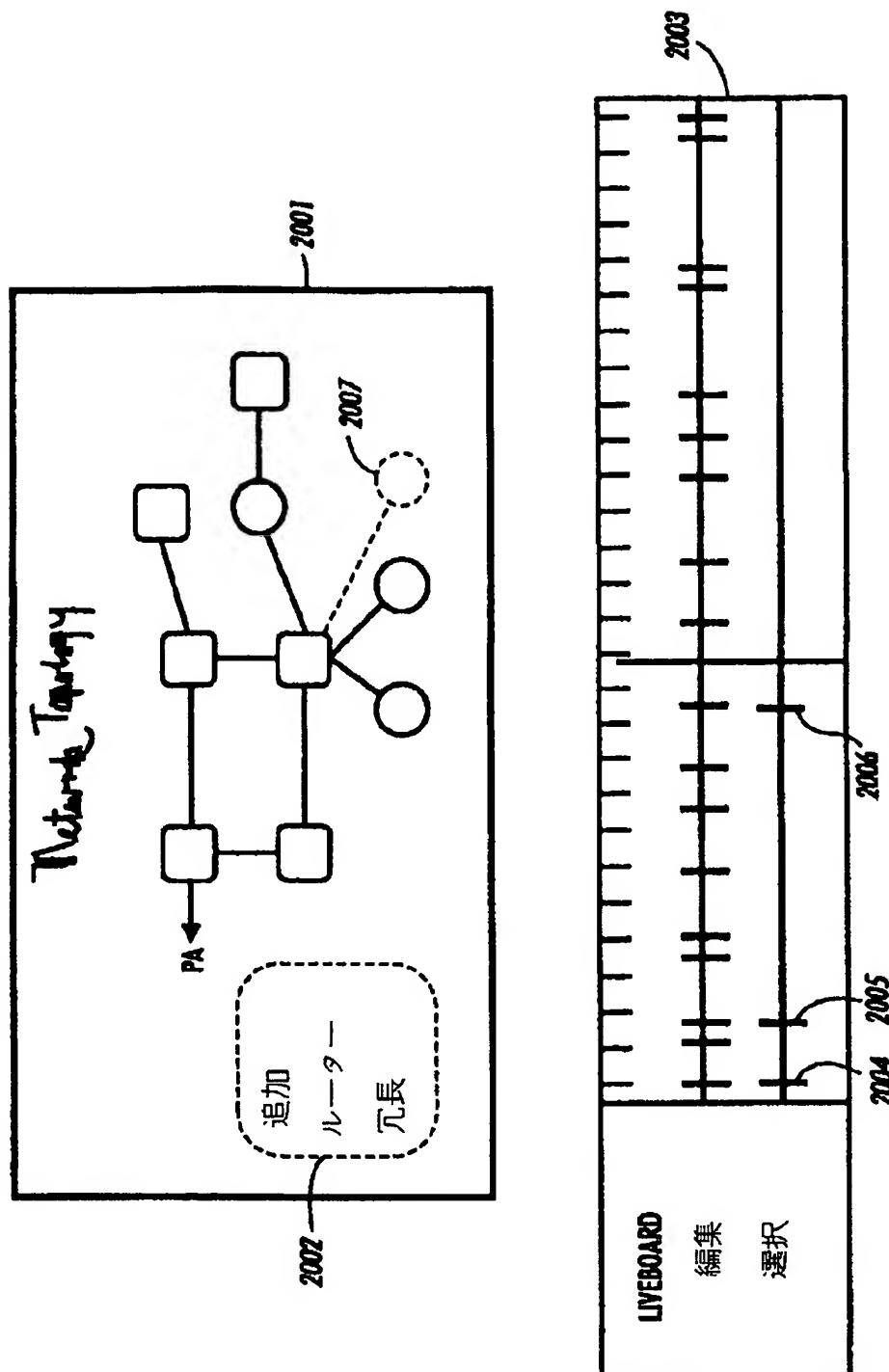
[Drawing 18]



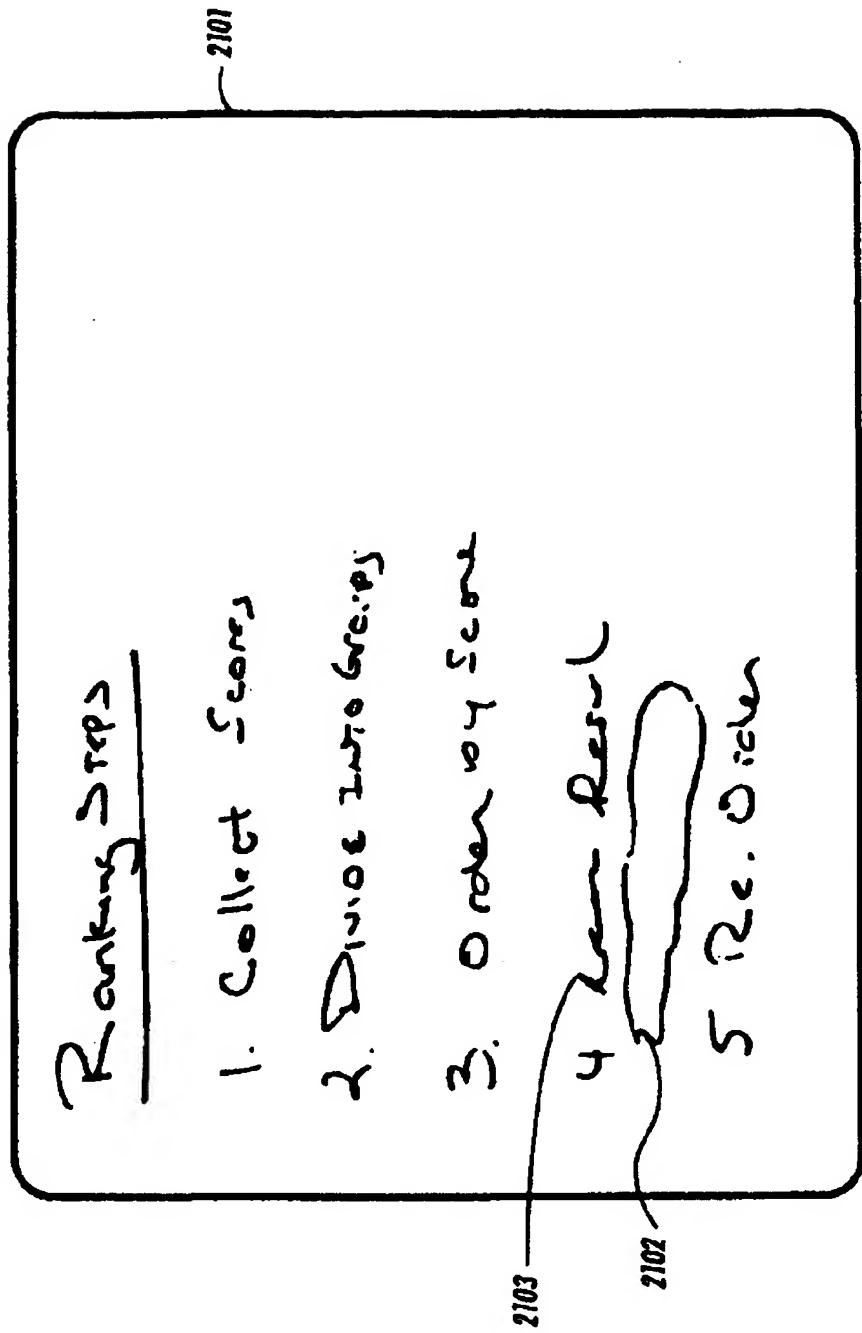
[Drawing 19]



[Drawing 20]



[Drawing 21]



[Drawing 22]

## Ranking Steps

1. Collect Scores
2. Divide into Groups
3. Order by Score
4. ~~Normalize Result~~
5. Re. Order

①

ランキングに  
関する議論

2202

結果を  
通常化

2303

2201

[Translation done.]